Semantic Relations-II.
Enhancing Resources and Applications

Workshop Programme

22 May 2012

09:00-10:30– Semantic Relations Identification and Extraction

Sara Mendes, Silvia Necşulescu, Núria Bel, *Synonym Extraction Using a Language Graph Model*

Pilar León Araúz and Pamela Faber, *Causality in the Specialized Domain of the Environment*

Scott Piao, Diana Bental, Jon Whittle, Ruth Aylett, Stephann Makri, Xu Sun, *A Pilot Study: Deriving a Users' Goal Framework from a Corpus of Interviews and Diaries*

10:30 – 11:00 Coffee break

11:00 – 13:00 Enhancing Applications

Agata Cybulska and Piek Vossen, *Using Semantic Relations to Solve Event Coreference in Text*

Gerold Schneider,*Using Semantic Resources to Improve a Syntactic Dependency Parser*

Darja Fišer, Polona Gantar, Simon Krek, *Using Explicitly and Implicitly Encoded Semantic Relations to map Slovene WordNet and Slovene Lexical Database*

Sruti Rallapalli and Soma Paul, *Evaluating Scope for Labeling Nominal Compounds Using Ontology*

13:00 – 14:00 Lunch break

14:00 – 16:00 Enhancing Resources

Daniela Katunar, Matea Srebáčić, Ida Raffaelli, Krešimir Šojat,*Arguments for Phrasal Verbs in Croatian and Their Influence on Semantic Relations in Croatian WordNet*

Silke Scheible and Sabine Schulte im Walde, *Designing a Database of GermaNet-based Semantic Relation Pairs Involving Coherent Mini-Networks*

Vasile Rus, Mihai Lintean, Cristian Moldovan, William Baggett, Nobal Niraula, Brent Morgan, *The SIMILAR Corpus: A Resource to Foster the Qualitative Understanding of Semantic Similarity of Texts*

Ştefan Daniel Dumitrescu, *Building a Baseline Supervised Relation Extraction System Using Freely-Available Resources*

Abhimanu Kumar, Richard Chatwin, Joydeep Ghosh, *Simple Unsupervised Topic Discovery for Attribute Extraction in SEM Tasks using WordNet*

16:00 – 16:30 Coffee break

16:30– 17:30 Invited talk: Patrick Hanks, *Mapping Semantic Relations onto Patterns of Word Use using Corpus Evidence*

Editors

Verginica Barbu Mititelu	RACAI
Octavian Popescu	FBK
Viktor Pekar	OUP

Workshop Organizers

Verginica Barbu Mititelu	RACAI
Octavian Popescu	FBK
Viktor Pekar	OUP

Workshop Programme Committee

Eduard Barbu	Universidad de Jaen
Antonio Branco	Faculdade de Ciências de Lisboa
Elena Cabrio	INRIA
Corina Forăscu	UAIC
Nuria Gala	LIF-CNRS
Patrick Hanks	UWE
Amaç Herdağdelen	Crimson Hexagon
Diana Inkpen	University of Ottawa
Radu Ion	RACAI
Elisabetta Jezek	Universita di Pavia
Svetla Koeva	IBL
Gerhard Kremer	Universität Heidelberg
Hristina Kukova	IBL
Claudia Kunze	Qualisys GmbH
Svetlozara Leseva	IBL
Bernanrdo Magnini	FBK
Emanuele Pianta	FBK
Reinhard Rapp	University of Leeds
Didier Schwab	Laboratoire d'Informatique de Grenoble
Carlo Strapparava	FBK
Sara Tonelli	FBK
Dan Tufiş	RACAI
Michael Zock	LIF-CNRS

Table of contents

Author Index

Preface

The present edition of the workshop *Semantic Relations* builds on the interest manifested by the participants to the first workshop *Semantic Relations. Theory and Applications* (held in conjunction with LREC2010) and by the large scientific community of linguists and language engineers.

At the first edition, we aimed at bringing together researchers in computational linguistics and lexical semantics, discussing theoretical and practical aspects of semantic relations and answering the question of how computational linguists could benefit from the work done by theoretical linguists and vice versa. In this second edition we focus on the benefits recources development and practical tasks in Natural Language Processing (NLP) have from and for the studies in lexical semantics.

The experience accumulated in Corpus Linguistics has shown that, while a large part of language use is regular and predictable, there is a significant part of it that is highly irregular and ambiguous. The research in this area suggests that lexical knowledge at large, encompassing all the information about lexical units, as well as the relationships between them, is instrumental in the accurate processing of natural language by computational methods.

There is a rising interest, both in theoretical and computational linguistics, in investigating the types of information that must be represented, and how these types could be conveniently organized in the lexicon in order to adequately describe the process of semantic interpretation. Specifically, the focus is on the relevant information which renders explicit the combinatorial process through which the meaning is formed – at a first level, the process of composing the lexical meaning from morphological parts, and, at a second level, the process of combining the conceptual knowledge of individual words, such as ontological categories and various relationships among them, into phrases carrying definite syntactic and semantic structures.

At a morphological level two research lines are noticeable. On the one hand, the study of affixes by means of which new words are created from existing ones sheds light on certain semantic relations between stem and their derived words; these relations are valid at a cross-lingual level, thus transferable among aligned resources and usable in various application. On the one hand, there is a growing interest in research into semantic relations either within compounds or between simplex and compounds, in their semantic representation with benefits for NLP tasks. The interest goes even beyond compounds and extends to set phrases and terms, while various domains are favoured, especially biomedicine. While theoretical linguistics establishes the possible relations involving compouns, computational linguistics automatically predicts and classifies these relations.

At a phrase level, there is an ongoing effort in NLP to automatically extract a large spectrum of semantic relations from various (semi)structured texts. From IS-A, part and causality to person-affiliation, organization-location and to abstract patterns encoding the relationships between words in conventional usage, the semantic relations have made their way into natural language processing. The properties of semantic relations are exploited in the economical design of language resources: (i) the transitivity of hyponymy relation, for example, is appropriate for nouns hierarchical organization based on inheritance properties of natural language, (ii) the patterning of verbs behavior shows that it is possible to represent the interconnection between lexical knowledge and world knowledge in a computable way.

The information required by automatic text processing using lexical-semantic relations can be acquired through corpus investigation methods and through data analysis in a strong sense. Accordingly, the lexicon could and should be built in a bottom-up manner by validating the phenomena mined from corpora by various computational methods.

In this edition of the workshop we wanted to highlight the interrelation between the quality and coverage of resources and the quality of applications relying on semantic relations.Specifically, in the call for papers we solicited papers on the following topics:

- Knowledge representation and semantic relations
- Extraction of semantic relations from various sources
- Exploitation of semantic relations in NLP applications
- Co-occurrence and semantic relations
- Lexical knowledge, world knowledge and semantic representation
- Patterns and semantic relations
- Semantic relations and word formation (compounding and derivation)
- Semantic relations and language learning and acquisition
- Semantic relations and language generation
- Semantic relations and terminology
- WordNets development

Most of these topics lie at the heart of the papers that were accepted to the workshop.

We would like to thank all the authors who submitted papers, as well as the members of the Program Committee for the time and effort they contributed in reviewing the papers. We are grateful to prof. Patrick Hanks for accepting to give an invited talk.

The Editors

Synonym extraction using a language graph model

Sara Mendes[a,b], Silvia Necşulescu[a], Núria Bel[a]

[a]Universitat Pompeu Fabra
Institut Universitari de Lingüística Aplicada
Roc Boronat 138, 08018 Barcelona, Spain

[b]Centro de Linguística da Universidade de Lisboa
Av. Prof. Gama Pinto, 2
1649-003 Lisboa, Portugal

E-mail: {sara.mendes, silvia.necsulescu, nuria.bel}@upf.edu

Abstract

One of the main requirements for lexical knowledge bases to be usable in NLP applications, apart from an appropriate data model, is a satisfactory level of coverage. Manually developed language resources are accurate, balanced and very reliable, but the cost of building them, both in terms of human resources and of time consumption has been a setback for their real application. Thus, conceiving methods for an automatic and fast development of language resources capable of providing adequate and reliable data for different languages and for different domains, if necessary, has become crucial in the field of NLP applications. The work presented here is framed by this general research effort. We aim at ultimately contributing to reduce the human effort required in the development of rich language resources in this work. We focus on the automatic acquisition of synonymy relations. We use a graph model and similarity measures based on the information encoded in the graph to extract lists of synonym pairs from corpus data, showing how semantic relations, specifically synonym relations, can be successfully extracted from corpus data in an automatic way.

Keywords: semantic induction, synonymy, graph models, language topology

1. Introduction

Lexical databases are invaluable sources of knowledge about words and their meanings, with numerous applications in areas like NLP, IR, and AI. Moreover, advances in Natural Language Processing (NLP) make apparent how crucial understanding and processing the information conveyed by natural language utterances is, particularly for a growing number of applications dealing with word sense disambiguation, anaphora resolution, information retrieval, machine translation, human-machine communication, among others.

One of the main requirements for lexical knowledge bases to be usable in such applications, apart from an appropriate data model, is a satisfactory level of coverage. This way, developing computational lexica with rich linguistic information – often for different languages and for different domains – is a requirement for real NLP applications. Given the crucial role played by rich language resources in this domain, their development has been a major concern for researchers in Computer Science and Computational Linguistics (Jing *et al.*, 2000; Wandmacher *et al.*, 2007, among many others). Presently, many systems are using manually collected language resources, such as electronic dictionaries and wordnets, that neither cover all languages, nor all possible application domains or the range of information required by specific applications. WordNet (Miller, 1990; Fellbaum, 1998) is the most well-known and most widely used lexical database for English processing, and is the fruit of over 20 years of manual work carried out at Princeton University.

Manually developed language resources are accurate, balanced and very reliable, but the cost of building them, both in terms of human resources and of time consumption has been a setback for their real application[1]. Thus, conceiving methods for an automatic and fast development of language resources capable of providing adequate and reliable data for different languages and for different domains, if necessary, has become crucial in the field of NLP applications. Many researchers have focused on the massive acquisition of lexical knowledge and semantic information as automatically as possible, generally using pre-existing structured lexical resources, which constitutes a problem for less resourced languages. The work presented here is framed by this general research effort, but we aim at developing an approach which essentially depends on the data, and does not depend on (or presupposes) pre-existing structured lexical resources. Ultimately, we aim at contributing to reduce the human effort required in the development of rich language resources. Here we present a resource-light approach for detecting semantic similarity in corpus data. Although our methodology is appropriate for extracting any semantic similar word pairs, here we focus on identifying synonym pairs, which are the most similar words in language and are therefore expected to stand out from the rest of the data in what regards semantic similarity. The work presented in this paper only addresses the case of synonym nouns, but we expect our approach to be straightforwardly applicable to other POS. Here we describe an experiment in which we use a graph model to extract synonym pairs from corpus data. We chose a closed domain – medicine – for developing the experiment presented in section 6, both for questions regarding the control of the results of the experiment and for showing how our approach can contribute to a more

[1] Most NLP applications require lexica including 20 000 to 60 000 word-forms (Dorr & Jones, 1996), while the average time needed to construct a lexical entry by hand can amount to as much as 30 minutes (see for instance Neff & McCord (1990) and Copestake *et al.* (1995)).

efficient development of rich language resources for specific domains. Specifically related to this last aspect, we will discuss how our approach allows for automatically pinpointing domain-specific synonym pairs (cf. section 3 for more details). In order to evaluate our approach, we use a POS-tagged English corpus, specifically the IULA medical domain corpus (Cabré *et al.*, 2006), containing 1.7 million tokens, so that we can compare our results against the synonymy relations listed in WordNet 3.1 (http://wordnet.princeton.edu/) and in the Merriam Webster Online (http://www.m-w.com/). In section 2 we present the main motivations for the work presented here, particularly with regard to the choice of synonymy as the object of our study, continuing, in section 3, with a general characterization of this particular semantic relation. In section 4 we make a brief description of previous research in this field, identifying the main aspects we aim at addressing in our work. In section 5 we describe the graph model we use, motivating its use for synonym extraction from corpus data. We depict the general design of our model and describe how we put it to work to extract synonym pairs. The methodology and the experiments run are presented in section 6, and results are discussed in section 7. Section 8 presents our final remarks and perspectives in terms of future work.

2. Motivation

Lexical-semantic relations are organizing principles of the lexicon. Widely discussed, characterized and classified in the literature (Jackendoff, 1983; Cruse, 1986; Miller, 1990; Pustejovsky, 1995; Vossen, 2002, among many others), lexical-semantic relations have been shown to be on the basis of different linguistic phenomena, which are exploited for various tasks in Computational Linguistics, such as information retrieval, information extraction, summarization, among others. Encoding semantic relations in language resources has thus become an asset with an important impact on the performance of NLP applications.

In this paper we focus on synonymy relations, specifically on automatically extracting them from corpus data. Our motivation lies not only in the fact that synonyms are important for addressing various issues in NLP, such as text summarization, question answering or text generation, but also because synonyms are the basic building blocks in the construction of concept-based resources such as wordnets and WordNet-like language resources, since in this kind of language resources each node represents a concept and is identified by all the lexicalizations of that concept, i.e. sets of synonyms. Specifically, automatically extracting pairs of synonyms can provide crucial information for pinpointing pairs of words which share the same meaning in a given domain, hence allowing for automatically extracting hypothetical synsets[2].

[2] A synset is a set of words with the same part-of-speech that can be interchanged in a certain context. For example, {*car; auto; automobile; machine; motorcar*} form a synset in WordNet 3.1 because they can be used to refer the same entity.

Widely discussed, defined and characterised in the literature, one of the paradigmatic aspects regarding synonymy is that, differently from what happens with other semantic relations, there are no straightforward linguistic cues for synonymy relations as synonym pairs hardly ever co-occur in the same utterance. Hence, research on designing new methods for automatically acquiring synonymy relations is not only pertinent, but crucial in order to reduce human intervention in the acquisition of language resources such as the ones mentioned above, without compromising their linguistic accuracy. In this work we show how we can take advantage of material and tools from corpus linguistics to accomplish this.

3. Synonymy

As defined in linguistic tradition, synonyms are words sharing the same meaning, synonymy being generally defined with regard to the impact the replacement of an expression with another in a sentence has in terms of its truth value. Using an informal definition, synonyms must comply with the following conditions:

(1) *A and B are synonyms iff replacing A with B or B with A never changes the truth value of the sentence in which they occur.*

Due to the economy principle in language, true synonymy is a very rare phenomenon. This is why in the context of concept-based resources synonymy is used in a weaker sense, bound to a given context. This is particularly relevant for our case study, as we are dealing with a specific domain, in which we expect to find some domain-specific synonym pairs and not others that are typically interchangeable in the common lexicon. In this weaker sense of synonymy as bound to a specific context, and according to Miller & Fellbaum (1990), two words are synonyms in a linguistic context C if replacing one with the other does not change the truth value of C. The following linguistic test can be used to decide whether two words are synonyms or not.

(2) if *A* and *B* are synonyms (in C)
then
(i) an *A* is a *B*
and
(ii) a *B* is an *A*

The symmetry of the relation, mirrored in the test, allows for distinguishing synonyms and pairs of words which satisfy the substitution test in some circumstances, such as hypernyms for instance.

(3) a. John parked the *car* in the back yard.
b. John parked the *vehicle* in the back yard.
c. If *A* is a car, then *A* is a vehicle. ***True***
d. If *B* is a vehicle, then *B* is a car. ***False***
Then: *A* is a *B* does not entail *B* is an *A*
=> *A* and *B* are **not synonyms**

e. John parked the *car* in the back yard.
f. John parked the *automobile* in the back yard.
c. If *A* is a car, then *A* is an automobile. ***True***
d. If *B* is an automobile, then *B* is a car. ***True***
Then: *A* is a *B* entails *B* is an *A*
=> *A* and *B* are **synonyms**

4. Acquiring synonymy relations

The most successful systems of lexical acquisition are based on the idea that the contexts in which words occur are associated to particular lexical types. Building from Harris' Distributional Hypothesis (Harris, 1951), a number of researchers (Fillmore, 1968; Grimshaw, 1990; Hearst, 1991; Levin, 1993, among others) have demonstrated conclusively that there is a clear relationship between syntactic context and word senses, a relationship that has been widely explored for the acquisition of semantic relations. Synonyms, by definition the most similar words in language in terms of semantic content, are expected to have similar contexts of occurrence.

Although different methods have been put forth by various researchers, most systems described in the literature work upon syntactic information as collected from a corpus and use different techniques to decide whether or not this information is relevant for classifying words. Different strategies have been put to work, particularly machine learning techniques using linguistic cues, with results comparable to those of state-of-the-art systems (Merlo & Stevenson, 2001; Baldwin & Bond, 2003; Baldwin, 2005; Bel *et al.*, 2007; Joanis *et al.*, 2007, among others). However, this type of approaches cannot be successfully used for our specific problem: exactly because of their equivalence in terms of semantic content, synonyms hardly co-occur in text, especially in the same sentence, leading to few indicative linguistic cues, if any.

Experiments involving the extraction of synonymy relations have been developed, particularly in the context of the automatic acquisition of wordnets and wordnet-like resources, as synonym sets are the basic building blocks for these resources. Consolidated as a standard *de facto* for the lexical-semantic representation of English, the Princeton WordNet is considered a reference and used in the development of new wordnets for other languages. In this context, different approaches have been put to work (Okumura & Hovy, 1994; Rigau *et al.*, 1995; Rigau & Agirre, 1995; de Melo, 2009; among many others), all of which have in common the fact that they exploit WordNet and bilingual resources like dictionaries and parallel corpora to obtain wordnets for other languages. Such work overcomes the question of time-consumption in the development of rich language resources, as it ensures that the huge networking effort needed to build a wordnet is done, but it has a few setbacks, both in terms of usability and linguistic accuracy of the resources obtained. Since it creates resources parallel-in-structure with WordNet, most of the criticisms pointed out to it also apply to the new resources automatically acquired in this way (senses are too fine-grained, lack of cross-POS relationships, simplicity of the relational information, etc.). Additionally, there is a methodological problem: the specific lexicalization patterns of individual languages are not mirrored in wordnets built this way, since what is obtained is the English structure with its nodes translated into other languages. Naturally, this is a bigger issue as the similarity between languages (i.e. from different language families and different cultural traditions, for instance) decreases.

In the context of the automatic acquisition of synsets there have also been experiments involving other kinds of resources. Oliveira & Gomes (2010), just to mention one, explore dictionary "definitions" – mostly definitions consisting of one word or an enumeration – to automatically obtain synsets for Portuguese. However, the results obtained lack in precision and linguistic accuracy, as very large synsets – too large for them to be usable in real NLP applications – are acquired: the average of variants per synset extracted in their experiment varies between 3,37 and 12,57, depending on the data sets, when the average in Princeton WordNet is 1,76.

Acquiring lexical-semantic structures is thus a hard problem and has been usually approached by reusing, merging and tuning pre-existing structured language resources, as in the research mentioned above. While in English the "lexical bottleneck" problem is softened to some extent (e.g. WordNet (Miller, 1990), Alvey Lexicon (Grover *et al.*, 1993), COMLEX (Grishman *et al.*, 1994) among many others), for other languages there are no wide range lexicons available. Our resource-light approach aims at overcoming this constraint. In the following sections we show how we extract synonym relations from corpus data, discussing our approach and the results obtained in an experiment.

5. The Graph Model

5.1 Motivation

The work presented here aims at reducing human intervention in the identification of pairs of synonyms from corpus data using a graph language model. This work is part of an ongoing research project on the topology of language particularly on the identification of semantic relations between words as emerging from language use. Graph language models are a suitable mathematical formalism to encode the relationship between words (Tsang *et al.*, 2010) and hence we will use it in the context of our work.

We build from Harris' Distributional Hypothesis (Harris, 1951) to automatically extract the lexical information needed for this directly from corpus data. Using real data as its source of information, our method naturally copes with language-dependent phenomena, hence overcoming one of the setbacks of the methods mentioned in the previous section.

Previous research in this area has focused on the vector space model in which the semantic similarity between two words is determined by calculating similarity measures between their vector representations (Grefenstette, 1994 ; Lin, 1998; Curran & Moens, 2002; Weeds, 2003; Turney *et al.*, 2010). Word frequency plays a very important role

in the vector space model approach, as the values encoded in the vectors are derived from the number of times a word occurs in a given context (Turney *et al.*, 2010). Also, all occurrences of a target word are considered when building vector elements, without taking into account phenomena such as word polysemy (Padó *et al.*, 2007), for instance. The semantic similarity between two words is thus calculated on the basis of all the contexts in which each word occurred and on their respective weights, which are derived from the frequency of that word and attached to every element in its vector. Given this, the vector space model is confronted with two major problems, which arise from general characteristics of human language: polysemy and the Zipfian distribution of words (Zipf, 1935).

Many researchers (Sahlgren, 2006; Peirsman *et al.*, 2008; van der Plas, 2008; Turney *et al.*, 2010) have stated that vector space models based on syntactic relations between words outperform co-occurrence models. However, this approach has an important drawback, as it involves external lexical resources, namely a parser that is very time-consuming, ambiguous and non-existing for many languages.

Our approach aims at tackling the problem of semantic similarity while avoiding the use of external lexical resources. We believe that semantically related words can be identified by using a graph language model and heuristic rules, hence avoiding the parser's drawbacks.

According to Zipf's law, the frequency of words is inversely proportional to its rank in the frequency table (Zipf, 1935), i.e. there will always be a large number of words that will appear very few times, if not only once, in any corpus of any length. The Zipfian distribution of words in particular is very relevant in the context of our work, as pairs of synonyms often show very different frequency in corpora, one of the elements of the pair being less used than the other. In fact, although synonyms are associated to an identical semantic content, denoting, in the case of nouns, the same set of entities in the world sometimes they are associated to different registers, one of the elements of the set of synonyms being a regionalism or an old-fashioned word, for instance. Being so, these different synonym words will be far from being equally represented in corpora.

In order to reduce the impact of word frequency in the identification of highly similar words in terms of their semantic content we rely on a directed graph model as our language topology. In the experiment presented in this paper we represent our corpus as a graph structure G=(V, E), where V={x_i} is the set of nodes in the graph and E={(x,y)| x, y ∈ V} is the set of arcs between two nodes. This means that in our work frequency is replaced by degree, i.e. we induce lexical information from the edges which link the nodes in our graph and not from the number of times they occur in our corpus. Below we present a detailed description of our approach.

5.2 Graph design

In the work presented here we focus on synonymy relations between nouns. We aim at extracting pairs of noun candidates and words representative of their semantic behaviour from the graph. Being so, low-frequent nouns cannot provide significant information for our task. Instead, in general, they cause sparseness in the graph, introducing noise as their low number of occurrences prevents any automatic system from deducing their semantic behaviour.

For the synonym detection task, the arcs in the graph should represent characteristic features of words semantic behaviour. Thus, bigrams that occur only by chance in corpus, i.e. that are not reinforced by reiterated uses, must be eliminated, as they are considered not to be significant for the semantic characterisation of the target word. This means that we only consider as candidate nouns the words whose occurrences in corpus give at least a minimal quantity of information about their behaviour in language. As our methodology is based on statistics, we use a frequency cut-off for bigrams. We collect directed word bigrams using a window of four that only contains words that are relevant for characterising the distribution of nouns: nouns, adjectives, verbs and prepositions[3], all other word classes not being considered, as they are not relevant for the task at hand. Words occurring in our corpus are lemmatised and then we discard bigrams that occur less than a threshold=10, as suggested in van der Plaas (2008). The remaining bigrams are considered word co-occurrences. The set of lemmas present in the final list of bigrams, disambiguated by their POS tag, constitutes the set of nodes in the graph. For each bigram (w_1, w_2) its corresponding arc [v_1, v_2] is added to the graph model.

5.3 Synonym extraction

According to the distributional hypothesis there is a strong relationship between the contexts in which a word occurs, i.e. its distribution, and its semantic content. Different techniques have been used in the literature to establish the context of occurrence of a word (cf. section 4). Using the graph described in the previous section, we aim at extracting a list of possible synonym pairs. The eligible synonym pairs are chosen from the set of pairs of any two words that share more than one context (P,S), i.e. two nodes that share at least one predecessor (P) and one successor (S), one word on each side being necessarily a content word, i.e. not a preposition. Being so, in the work presented here we consider as the context of occurrence of a word any pair (P,S) where:

(4) P={p∈ V| (p,w)∈ E}, |P|>=1, ∃p∈ P,
POS(p)!=preposition, ∀p_i, ∃p_j ∈ V, (p_i,p_j) ∈ E}
S={s∈ V| (w,s)∈ E}, |S|>=1, ∃s∈ S,
POS(s)!=preposition; ∀s_i, ∃s_j ∈ V, (s_i,s_j) ∈ E}

In our experiments, depicted below, we test several

[3] Very general prepositions, with an essentially "grammatical" content ("to", "at", "for", "on", "in", "by", "with" and "of"), are not included in our graph, as they tend to combine with any word in the corpus, thus not being distinctive for the sake of the task at hand.

variations of this model in which we apply different definitions of word context. We use weaker and stricter definitions of word context, the first defined by the conditions described above, and the second by adding the following constraint: considering that any two nodes are eligible synonyms if they co-occurred with the same preceding and following words in a window of four, we introduce the condition that each predecessor/successor and the target pair (the target word and its candidate synonym) share another predecessor/successor, this way assuring that they co-occurred in the same context in corpus, thus filtering the amount of context considered and hence hoping to consider more reliable information for the sake of our task.

Given we aim at reducing human intervention to a minimum in the extraction of synonyms from text, we try to provide the user with a list of possible synonym pairs as reduced as possible, and with a minimum amount of noise. Hence, if two words do not share enough context in the corpus, we do not consider them to be eligible pairs. By enough context we mean any pair (P,S) where $|P|*|S| \geq 10$. Below we discuss the soundness of introducing this threshold, as empirical data comes to support this option. This way, balancing the amount of constraints considered and the reduction in the number of contexts extracted that results from it, and thus, the reduction in the amount of information available to our system, is crucial in the context of our work.

Moreover, considering that, as aforementioned, synonyms are words that hardly ever co-occur in the same utterance, we discard all pairs of words that are neighbours in the graph from our list of possible synonyms.

Finally, we use the common context shared by the word pairs in our list to estimate the semantic relationship between them. We calculate the semantic similarity between word contexts using Dice's coefficient:

(5)

$$\frac{2*|\text{Significance}(\text{Context}(X) \cap \text{Context}(Y))|}{|\text{Significance}(\text{Context}(X)) \cup \text{Significance}(\text{Context}(Y))|}$$

We tested two models to compute context significance: one in which we consider the weight of the arcs; and another in which we do not. The weights give more importance to less expected events and less importance to expected events. In the context of our work, a larger importance is thus given to words with less edges in the graph model, as we expect them to be more informative with regard to the semantic characterisation of a target word. Therefore, we weight the significance of each arc using the inverse of its degree. This way we reduce the impact of word frequency in our results as we take into account the number of context features for each word instead of its frequency in corpus. Therefore, if w_i and w_j share the context (P,S):

(6)

$$\text{significance}(\text{context}(P,S)) = \sum_{p \in P} \frac{1}{\text{degree}(p)} * \sum_{s \in S} \frac{1}{\text{degree}(s)}$$

In the following sections we depict our experiments and results, discussing how we come to filter the data extracted as described in this section in order to reduce the noise in the results as much as possible without loosing accurately extracted synonym pairs in the process.

6. Experiments and Methodology

For our synonymy extraction experiment we used our graph language model (cf. section 5 for a detailed description on the motivations and design of the graph model used in this work) and the semantic similarity measures described in section 5.3. The baseline has been the list of candidate synonym pairs fulfilling the conditions of the weaker definition of word context in (4). Our experiment aimed at showing to what extent the distributional information extracted from our graph, in combination with semantic similarity measures, was able to accurately rank the candidate synonym pairs, putting actual synonyms on top of the list, hence distinguishing them from other candidate pairs for which a synonymy relation does not hold. To evaluate our results we used a manually annotated list of synonyms as our gold-standard, consisting of all the word pairs occurring in our graph and listed as synonyms either on WordNet 3.1 (http://wordnet.princeton.edu/) or Merriam Webster Online (http://www.m-w.com/).

Our baseline contains 935 candidate synonym pairs extracted from our graph which have more than 10 common contexts. Moreover, looking at the data, we realize that there is a small set of nodes in our graph that apparently function as hubs, displaying arcs with almost every other node in the graph and hence not introducing distinctive information for a task such as ours. In order to evaluate to which extent these nodes have an exceptional distribution when compared against the other nodes in the graph, we calculated the mean and standard deviation (σ) of the degree of the graph nodes to establish the dispersion of this measure in our data, and considered the nodes with a degree higher than the mean+2σ to be outliers. We found the following set of outlier nodes: the nouns *cell*, *gene* and *protein*; the adjective *human*; and the verb *use*. Analyzing the set of possible synonyms in our baseline, 262 pairs were identified as having at least either the predecessor or successor list of contexts containing only outlier nodes. As these nodes do not provide distinctive information regarding the distributional behaviour of candidate words, we consider these candidate pairs not to have reliable enough contexts to be declared synonyms. Also, an important number of candidate pairs contains more outliers than reliable nodes in their common context, hence introducing a considerable amount of noise in our candidates list. Being so, we considered important to evaluate the influence of this set of outlier nodes in the semantic similarity measure.

In order to do so, we performed four different tests: one in which we eliminated the set of outlier nouns from the list of common context of our candidate pairs; another in which we eliminated the adjective *human*; a third one in which we removed the verb *use* from our graph; and a

final one in which we eliminated all the outliers.

We used different strategies to discard the outlier nouns and adjective, on the one hand, and the outlier verb on the other. While outlier nouns and adjective were simply removed from the list of contexts of each word, the verb removal was slightly more complex. As verbs generally put two or more nouns in relation, we addressed the outlier verb *use* as a function word, and added an edge between its predecessor (typically the subject) and its successor (typically the object) as we removed it from our graph, relying on the idea that there is a semantic relation between the subject and the object of a given verb which can provide relevant information for classifying words, and specifically for identifying synonym pairs.

Eliminating the contexts involving outlier nouns eliminates 439 candidates from our list, while removing the adjective *human* only reduces the list with 61 pairs. Removing the verb *use* from our graph is also not very significant in filtering noise, resulting in a reduction of only 69 word pairs. But removing all the outlier nodes at once results in a considerable pruning of our list, leaving us with a list of 311 candidate pairs, which corresponds to a 65% reduction with regard to the baseline. Naturally, the number of candidate pairs is not the only thing being reduced with this strategy. The candidate pairs common context was also reduced in 42% after all the outliers were eliminated.

Our test set contains 44 synonym pairs manually identified, as described above, 16 of which have at least 10 common contexts in our baseline. As aforementioned, by eliminating the outlier contexts, we also reduce the number of contexts characterising synonym pairs correctly included in our lists. Our expectation was that synonym pairs would share enough distinctive context to emerge and distinguish themselves from other pairs of candidate words. As shown below, that is indeed what happens, except in the case of 9 synonym pairs in our gold standard which have their shared contexts eliminated to zero. Apparently an undesirable effect of the outlier filter, it must be underlined that this essentially happens to pairs of synonyms sharing less than 10 contexts in our baseline, hence providicommon evidence that the $|P|*|S| \geq 10$ condition we are using is in fact motivated, as less than this amount of contexts is clearly not enough context for any classification to be made conclusively. This means that our corpus does not provide enough information about these word pairs for synonymy to be identified automatically. For instance once the outlier nodes are eliminated from our graph we lose the "*function-role*" synonym pair, as all their common successors were outlier nodes {J=[human/JA], N=[gene/N5, cell/N5, protein/N5]}, exactly the same successors shared by the word pair "*region-role*", between which a synonymy relation does not exist, the common predecessors not being sufficient to distinguish between the two word pairs.

As we are aiming at developing a semi-supervised system, our goal consisting in reducing the time invested in synonym detection by human developers to a minimum, we are more focused on precision, than on recall. Moreover, 8 out of these 9 pairs which have their common contexts reduced to zero, would have been eliminated by the $|P|*|S| \geq 10$ condition anyway.

	baseline	no noun outliers	no adjective outliers	no verb outliers	no outliers
candidates	935	496	874	866	311
syn detected	16	13	15	15	13
syn ranked in first 100 pairs	10	11	11	11	12
syn ranked higher than baseline		12	13	15	12
syn ranked lower than baseline		1	2	0	1
syn lost		3	1	1	3

Table 1: Baseline results compared against the system results when outlier nodes are eliminated.

In table 1 above we compile the data described above, placing the information on the amount of candidate synonyms extracted in each case side by side, and adding information on the ranking in which correctly extracted synonyms appear in the list of candidates. Removing the outlier nodes from our results has a crucial impact in the reduction of candidate pairs, without generating too many false negatives, while simultaneously promoting actual synonyms in the list, moving them higher in almost every case.

Once the candidate list has been reduced and an important part of the noise eliminated by our outlier nodes filter, our main focus becomes ranking synonym pairs at the top of the list.

We applied two additional constraints to our results to evaluate their impact in the ranking of actual synonyms across the list of candidates:

1. weighting the graph arcs taking into account the degree of the nodes they connect, as presented in (6);
2. considering a stricter definition of word context, as described in section 5.3.

We tested the individual contribution of each additional constraint in the performance of our system, and finally the impact of combining them.

In table 2 we present the impact this additional constraints have in the performance of our system, once again by putting in parallel the total number of synonyms correctly identified in each test run.

Finally, in figure 1 we present five dispersion graphics for the five experiments referred in table 2. These represent the ranks in which correctly identified synonym pairs appear in the lists extracted in our experiments.

	baseline	no outliers unweighted	no outliers weighted	no outliers unweighted with filter	no outliers weighted with filter
candidates	935	311	311	262	262
syn detected	16	13	13	9	9
syn ranked in first 100 pairs	10	12	10	9	7
syn ranked higher than baseline		12	11	9	6
syn ranked lower than baseline		1	2	0	3
syn lost		3	3	7	7

Table 2: Baseline results compared against the system results when outlier nodes are eliminated and when we introduce weight in the similarity measure.

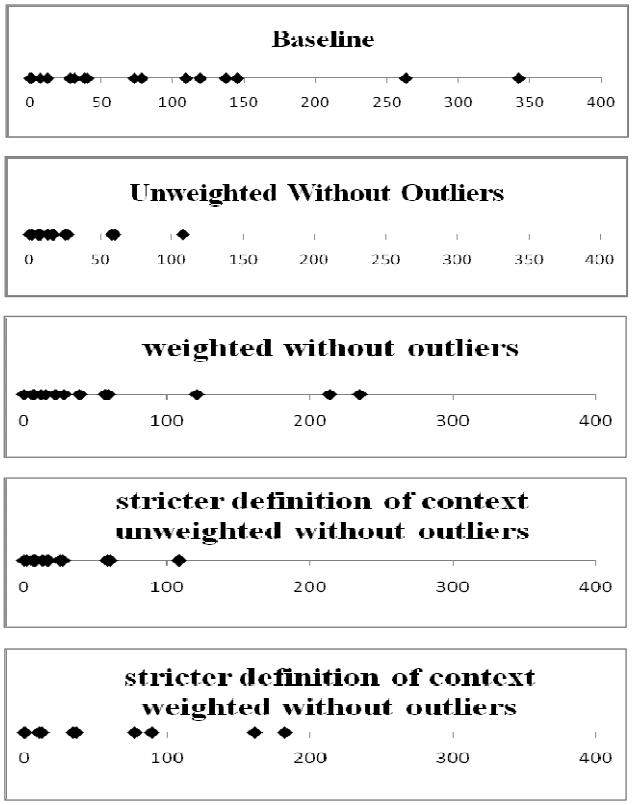


Figure 1: Baseline results compared against the system results when outlier nodes are eliminated and when we introduce weight in the similarity measure and when we apply the stricter definition of context.

In these graphics it becomes apparent to which extent actual synonym pairs are promoted in the list of candidate pairs, concentrating in top positions when outlier nodes are eliminated from our graph, and being slightly more dispersed when weights are considered in our similarity measure. Also, by looking at the information in table 2, we realize that when a stronger constraint is applied over the context definition there is a significant number of synonyms lost along with the pruning of candidate pairs. We discuss these results in more detail in section 7.

7. Results and discussion

From the experiments described in section 6, two apparently contradictory observations can be made. On the one hand, it becomes very clear that when there is more context information regarding a pair of candidate words, its classification is more accurate and stable. On the other hand, this amount of context information is only significant for our task when it is distinctive, i.e. context information that is shared by a very large part of the graph nodes can and must be discarded, as it does not provide distinctive information about words and introduces noise. As presented in section 6, when we discard the contexts involving outlier nodes in our graph with regard to their degree, i.e. nodes that function as hubs in our graph, we filter an important part of the noise in our candidate synonym lists and promote actual synonym pairs in the ranking provided by the similarity measure between candidate words. As discussed above, that is indeed what happens, except in the case of 9 synonym pairs in our gold standard which have their shared contexts eliminated to zero, which is apparently an undesirable effect of our filter. Besides the observations regarding the minimum context constraint, modelled in our system by the $|P|*|S| \geq 10$ condition, presented in detail in section 6, another interesting observation regarding this amounts to the fact that there are synonym pairs discarded in this way that are in fact synonyms in other domains. Just to mention an example, one of the synonym pairs that is filtered out when we introduce the outlier restriction is the "*part-role*" candidate pair. It corresponds in fact to a pair of synonyms, but in the acting domain, generally not being used in the medical domain. This way, considering that our task aimed at identifying synonym pairs in the medical domain, discarding this word pair as a synonym in this domain ends up being linguistically accurate, and thus a desirable effect of our system.

Also, it becomes very clear from our experiments that introducing the weigh of the arcs in our graph in the similarity measure has a negative impact on the results. This corroborates the original intuition that led us to develop the graph model in the first place: frequency introduces a bias that does not allow for observing and extracting information from less represented facts in corpus data. Hence, as empirically shown by our experiment, it is more significant – since it has better overall results – to consider the existence of an arc between two words than to give a specific weight to that arc depending on the amount of relations the predecessor and/or the successor of the target word has.

Finally, using a stricter definition of context ends up ruling out a considerable amount of sound synonym pairs. This is due to the fact that applying this definition results in a considerable reduction of the amount of contextual

information available. Hence in future work we will further test this stricter definition of context using a larger corpus to evaluate to which extent it can contribute to enhance the results of our system when there are larger data sets available. Moreover, these results also make apparent how crucial it is balancing the amount of constraints considered to prune the results and reduce noise, and the reduction in the amount of information available to the system, which can compromise its performance.

To conclude our discussion of the results, a coarse analysis of the noise that persists in our lists is in order. As indicated in the tables of results presented in the previous section, there is still a considerable amount of candidate pairs in our lists that are not synonyms, but are semantically related words which are related by other types of semantic relation, such as part-whole relations or hypernymy-hyponymy. This is an expected result of our system, as it is based on the induction of semantic similarity from corpus data. One of the relations which is also considerably represented in the top ranks of our lists is antonymy. Considering that antonym words are also characterised by having a similar linguistic distribution, i.e. by occurring in the same linguistic contexts, this is also an expected effect of our approach and, thus, evidence that our system is working as it was supposed to. Having these two aspects in mind, we will pursue these lines of research, aiming at filtering out these word pairs from our lists. Some general lines on possible approaches to this question are discussed below, in our final remarks.

8. Final remarks

The work described in this paper is part of ongoing research on the topology of language and on language structure as emerging from language use. Regarding the task of synonym extraction, particularly considering its complexity and the challenges it involves (ambiguity problems, domain dependence, sparseness of corpora data for some of the candidate words, lack of linguistic cues for the identification of synonymy relations, just to mention the most relevant ones), the results obtained in our experiments not only are promising, but seem to leave room for further improvement.

As future work we will evaluate the possibility of further filtering the lists of candidate synonyms extracted from our graph by working on the identification of other semantic relations, particularly those for which linguistic cues can be straightforwardly found in corpus data. We will look into the possibility of combining the results of such classifiers with the language graph model described in this paper, thus eliminating semantic relations other than synonymy from our results, hence reducing a considerable part of the noise that still persists in our results.

We must underline, however, that the results obtained in our experiments are suitable for the goal we were set to pursue: the semi-automatic construction of rich language resources. In fact, considering the filtered amount of synonym candidates and the ranking of actual synonyms in our lists, as provided by the system, identifying them is a very straightforward task for human developers, which moreover can be accomplished in a reduced amount of time. This way, the work described here can contribute to viable and efficient strategies in the development of language resources that still assure their accuracy, coverage, volume and usability.

9. Acknowledgements

We thank Muntsa Padró for her valuable suggestions and comments in the development of this work. We also wish to thank the remarks of the anonymous reviewers that contributed to the final version of this paper.

Sara Mendes and Silvia Necşulecu's work presented in this paper has been supported respectively by Fundação para a Ciência e a Tecnologia post-doctoral fellowship SFRH/BPD/79900/2011 and by the CLARA project (EU-7FP-ITN-238405).

10. References

Baldwin, T. (2005). "General-Purpose Lexical Acquisition: Procedures, Questions and Results". *Proceedings of the Pacific Association for Computational Linguistics 2005*, Tokyo, Japan.

Baldwin, T. & F. Bond (2003). "Learning the Countability of English Nouns from Corpus Data". *Proceedings of the 41st. Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan.

Bel, N., S. Espeja & M. Marimon (2007). "Automatic Acquisition of Grammatical Types for Nouns". *Human Language Technologies 2007: The Conference of the NAACL, Companion Volume, Short Papers*. Rochester, USA, pp. 5-8.

Cabré, M. T., Bach, C., and Vivaldi, J. (2006). 10 anys del Corpus de l'IULA. Barcelona: Institut Universitari de Linguística Aplicada. Universitat Pompeu Fabra.

Copestake, A., T. Briscoe, P. Vossen, A. Ageno, I. Castellon, F. Ribas, G. Rigau, H. Rodríguez & A. Samiotou (1995). "Acquisition of Lexical Translation Relations from MRDS". *Machine Translation*, 9.

Cruse, D. (1986). *Lexical Semantics*. Cambridge: Cambridge University Press.

Curran, J. R. & M. Moens (2002). "Improvements in automatic thesaurus extraction". *Proceedings of the Workshop on Unsupervised Lexical Acquisition*.

de Melo, G. & G. Weikum (2009) "Towards a Universal Wordnet by Learning from Combined Evidence". *Proceedings of 18th ACM Conference on Information and Knowledge Management (CIKM 2009)*. Hong Kong, China.

Dorr, J. & D. Jones (1996). "Acquisition of Semantic Lexicons: using word sense disambiguation to improve precision". *Proceedings of the SIGLEX Workshop "Breadth and Depth of Semantic Lexicons"*. Santa Cruz, California, USA, pp. 42-50.

Fellbaum, C. (1998). "A Semantic Network of English: the Mother of all WordNets". In Vossen, P. (ed.), *EuroWordNet: a Multiligual Database with Lexical Semantic Networks*. Dordrecht: Kluwer Academic

Publishers, pp. 137–148.

Fillmore, C. J. (1968). “The Case for Case”. In Bach, E. & R. T. Harms (eds.), *Universals in Linguistic Theory*. Holt, Rinehart, and Winston, pp. 1-88.

Grefenstette, G. (1994). *Explorations in automatic thesaurus discovery*. Norwell, MA, USA: Kluwer Academic Publishers.

Grimshaw, J. (1990). *Argument Structure*. Cambridge: The MIT Press.

Grishman, R., C. Macleod & A. Meyers (1994). “Comlex syntax: building a computational lexicon”. *Proceedings of the 15th Annual Meeting of the Association for Computational Linguistics (Coling'94)*. Kyoto, Japan, pp. 268-272.

Grover, C., J. Carroll & J. Reckers (1993). “The Alvey Natural Language Tools grammar (4th realese)”. *Technical Report 284*. Computer Laboratory, Cambridge University, UK.

Harris, Z. (1951). *Structural Linguistics*. Chicago University Press.

Hearst, M. (1991). “Noun Homograph Disambiguation Using Local Context in Large Text Corpora”. *Proceedings of the 7th Annual Conference of the University of Waterloo Centre for the New OED and Text Research*. Oxford, England, pp. 1-22.

Jackendoff, R. S. (1983). *Semantics and Cognition*. Cambridge, Massachussets: The MIT Press

Jing, H., Y. D. Netzer, M. Elhadad & K. McKeown (2000). "Integrating a large-scale, reusable lexicon with a natural language generator". *Proceedings of the 1st International Conference on Natural Language Generation* . Mitzpe Ramon, Israel.

Joanis, E., S. Stevenson & D. James (2007). “A General Feature Space for Automatic Verb Classification”. *Natural Language Engineering*, 14.

Levin, B. (1993). *English Verb Classes and Alternations: a preliminary investigation*. Chicago, IL: The Chicago University Press.

Lin, D. (1998). “An information-theoretic definition of similarity”. *Proceedings of International Conference on Machine Learning*. WI: Madison.

Merlo P. & S. Stevenson (2001). “Automatic Verb Classification based on Statistical Distribution of Argument Structure”. *Computational Linguistics*, 27:3.

Miller, G. A. (1990). “WordNet: an online Lexical Database”. *Special Issue of International Journal of Lexicography*, vol. 3, nº4.

Neff, M. & M. McCord (1990). “Acquiring Lexical Data from Machine-Readable Dictionary Resources for Machine Translation”. *Third International Conference on Theoretical an Methodological Issues in Machine Translation of Natural Language (TMI-90)*. Austin, Texas.

Okumura, A. & E. Hovy (1994) “Building Japanese-English dictionary based on ontology for machine translation”. *Proceedings of the Workshop on Human Language Technology, HLT*. Morristown, NJ, USA: Association for Computational Linguistics, pp. 141–146

Oliveira, H. & P. Gomes (2010). “Towards the automatic creation of a wordnet from a term-based lexical network”. *Proceedings of the ACL Workshop TextGraphs-5: Graph-based Methods for Natural Language Processing*.

Padó, S., & M. Lapata (2007). “Dependency-Based Construction of Semantic Space Models”, *Computational Linguistics 33(2),* pp. 161-199.

Pustejovsky, J. (1995). *The Generative Lexicon.* Cambridge, MA: The MIT Press.

Peirsman, Y., K. Heylen & D. Speelman (2008). “Putting things in order.First and second order context models for the calculation of semantic similarity”. *Actes des 9es Journées internationales d'Analyse statistique des Données textuelles (JADT 2008).* Lyon: France.

Rigau, G. & E. Agirre (1995). “Disambiguating bilingual nominal entries against WordNet”. *Proceedings of The Computational Lexicon Workshop at the Seventh European Summer School in Logic, Language and Information (ESSLLI ‘95)*. Barcelona, Spain, pp 71-82.

Rigau, G.., H. Rodríguez & J. Turmo (1995). “Automatically extracting translation links using a wide coverage semantic taxonomy”. *Proceedings of the fifteenth International Conference AI’95. Language Engineering ’95*. Montpellier, France.

Sahlgren, M. (2006). *The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. PhD Thesis.

Turney, P. D. & P. Pantel (2010). “From Frequency to Meaning:Vector Space Models of Semantics”. *Journal of Artificial Intelligence Research* 37 , pp. 141-188.

van der Plas, L. (2008). *Automatic lexico-semantic acquisition for question answering*. PhD Thesis, University of Groningen

Vossen, P. (2002). *EuroWordNet General Document*. EuroWordNet Project LE2-4003 & LE4-8328 report. University of Amsterdam.

Wandmacher, T., E. Ovchinnikova, U. Krumnack & H. Dittmann (2007). “Extraction, evaluation and integration of lexical-semantic relations for the automated construction of a lexical ontology”. *Proceedings of the Third Australasian Ontology Workshop (AOW 2007)*, Gold Coast, Australia. CRPIT, 85. Meyer, T. & A. C. Nayak (eds.), ACS, pp. 61-69.

Weeds, J. (2003). *Measures and Applications of Lexical Distributional Similarity*. PhD thesis, University of Sussex.

Zipf, G. K. (1935). *The Psychobiology of Language*. Houghton-Mifflin.

Causality in the Specialized Domain of the Environment

Pilar León Araúz and Pamela Faber
Department of Translation and Interpreting, University of Granada
Buensuceso, 11 18002 Granada
E-mail: pleon@ugr.es, pfaber@ugr.es

Abstract

EcoLexicon is a multilingual terminological knowledge base (TKB) that represents environmental concepts and their relations in different formats. In this paper we show how some of the manual processes that we have developed for the extraction and representation of semantic relations can be partially automatized with the help of NLP applications such as NooJ. Focusing on the causal relation, we have designed various graph-based micro-grammars to match and annotate the corpus. This permits the extraction of causal propositions, and identifies the terms that primarily act as causes and effects in environmental contexts. Finally, these grammars can also be used to measure the prototypicality of causal propositions within four different environmental domains.

Keywords: knowledge extraction, causal relations, semantic prototypicality, environment

1. Introduction

EcoLexicon[1] is a multilingual terminological knowledge base (TKB) that represents environmental concepts and their relations in different formats (i.e. ontology, conceptual networks, controlled-language definitions, graphical resources and linguistic contexts, such as knowledeg-rich contexts and concordances). So far, it has 3,343 concepts and 17,413 terms in English, Spanish, German, French, Modern Greek, Russian and Dutch. In this paper we show how some of the manual processes that we have developed in the extraction and representation of semantic relations can be partially automatized with the help of NLP applications such as NooJ (Silberztein, 2003).

2. Semantic relations in EcoLexicon

In addition to hyponymic relations, our inventory of semantic relations also includes six types of meronymy as well as non-hierarchical relations, such as *affects*, *result_of*, *causes*, etc., which best represent the dynamism of the environmental domain (León Araúz and Faber, 2010). Up to the present, all conceptual propositions in EcoLexicon (more than 6,000) have been manually extracted from the corpus (5 million words) and represented in semantic networks. However, knowledge representation would be more objective and efficient if knowledge extraction techniques were more systematic and semi-automatized. Nevertheless, this requires a well-defined set of selection criteria, based on the manual identification of which types of information are useful, why they are useful and how they can be structured.

2.1 Extracting semantic relations from the corpus

According to many research studies, knowledge patterns (KPs) have long been considered one of the most reliable methods for the extraction of semantic relations (Condamines, 2002; Marshman et al., 2002; Barrière, 2004; Barrière and Abago, 2006; Cimiano and Staab, 2006). The term KP was coined by Meyer (2001) to refer to the lexico-syntactic patterns between the terms encoded in a proposition in real texts.

Since Hearst (1992) much as has been written about KPs. Nevertheless, despite their popularity, KPs have never been fully studied and exploited. As Bowker (2004) states, there are still major problems with regards to noise and silence, pattern variation, anaphora, domain and language dependency, etc. Moreover, not all relations have been analyzed in the same depth. Patterns conveying hyponymic relations are the most commonly studied since they play an important role in categorization and property inheritance (Barrière, 2004: 244). Nonetheless, even though non-hierarchical KPs have also been identified by many other authors, they have never been systematically implemented in research studies (Aussenac-Gilles, 2000: 181).

KPs have mostly been used to extract information from general language texts, but they have also been applied in certain specialized domains, such as Medicine (Rosario and Hearst, 2004; Vintar and Buitelaar, 2003; Embarek and Ferret, 2008; Khoo et al., 1999) or Biopharmaceutics (Marshman, 2002). However, to the best of our knowledge, there have been no KP-related studies on the environment.

All approaches seem to agree that the use of KPs for knowledge extraction involves a series of complementary steps. Nevertheless, the order of the steps differs, depending on research objectives (e.g. identification of term pairs, discovery of new KPs, searching for known KPs to discover new term pairs, etc.). In Terminology, Meyer (2001) suggests first identifying an initial set of KPs for each semantic relation. These patterns are then tested and additional patterns are identified. Restrictions are subsequently defined that can be applied to reduce noise and silence. As part of our study, all of this has first

[1] http://ecolexicon.ugr.es

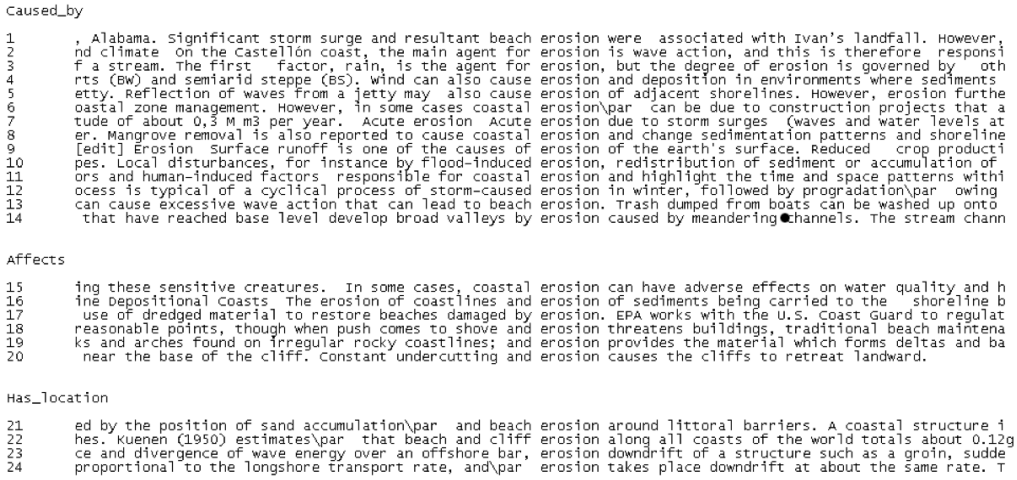


Figure 1: *Erosion* concordances

been done by manual corpus analysis.

For example, in Figure 1 we show the results of the first step in our approach. We search for specialized terms, such as *erosion*, collect the most meaningful concordances and classify them based on the relations expressed. KPs are then collected, such as those found in Figure 1: *associated with*, *agent for*, *can/may also cause*, *can be due to*, *one of the causes of*, *responsible for*, *lead to*, etc. The next step involves reusing these KPs to discover new term pairs, after which we reinitiate the process with seed terms to discover new KPs. This information is also displayed to users since those who are translators and/or technical writers might find it useful.

During the manual identification of KPs, we encountered certain problems related to the polysemic nature of certain KPs, which did not always convey the same semantic relations (i.e. *formed by*, León Araúz and Reimerink, 2010) or the problem of KPs associated with an incomplete proposition because of anaphora. Nevertheless, we also found that the correct identification of meaningful concordances depends on the semantic and syntactic structure of the text that precedes and follows any KP.

2.2 Representing semantic relations in conceptual networks

The semantic relations between concepts in EcoLexicon are activated depending on the *natural constraints* imposed by a concept's intrinsic nature and its relational power. The activation of relations also depends on the *contextual constraints* stemming from facet incompatibility, which is the result of multidimensionality (see León Araúz and Faber, 2010 for a more detailed explanation). Succinctly put, depending on the type of concepts in a conceptual proposition, only a certain set of relations may apply. For instance, a PHYSICAL ENTITY can only be the *result of* a PROCESS, but not of another ENTITY, and only if the PHYSICAL ENTITY plays the role of PATIENT and not that of AGENT. Furthermore, concepts in the environmental domain have multiple dimensions that are often incompatible because they are context-dependent. For example, despite that WATER is included in propositions such as <CONCRETE *made_of* WATER> and <WATER *causes* EROSION>, these propositions should evidently not be included in the same semantic network. Thus, even though a concept may be part of multiple propositions, only one set of these propositions should be activated in a certain context. Therefore, we have divided the environmental domain into field-specific contextual subdomains, such as HYDROLOGY, GEOLOGY, OCEANOGRAPHY, SOIL SCIENCES, ATMOSPHERIC SCIENCES, etc.

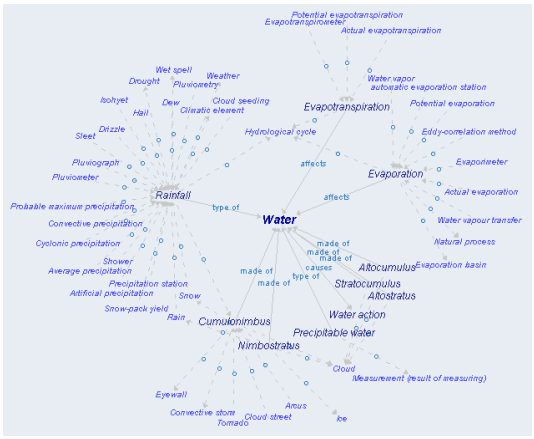


Figure 2: WATER in ATMOSPHERIC SCIENCES

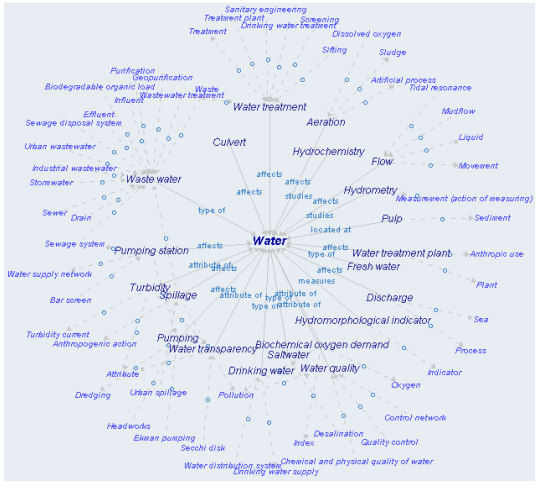


Figure 3: WATER in WATER TREATMENT

Each of these domains provides a frame for conceptual recontextualization. A comprehensive list of all contextual domains can be found in León Araúz and San Martín (in press). Figures 2 and 3, show the different recontextualizations of the semantic networks for WATER in the subdomains of ATMOSPHERIC SCIENCES and WATER TREATMENT. As can be observed, prototypical propositions for WATER (e.g. <WATER *causes* EROSION>), which would generally be activated in a context-free search, do not appear in either network. Instead, it is the context that modulates the prototypicality of propositions. The recontextualization of concepts thus involves decisions about which propositions should be activated within each domain. In EcoLexicon, so far, this has been done manually and intuitively, based on corpus searches and analysis. This time-consuming process has been extremely worthwhile in that it has provided us with the knowledge needed to formalize the structure of KPs for automatic corpus searches and determine the prototypicality of conceptual propositions. Accordingly, the corpus texts are currently being classified in contextual domains.

3. Causal relation

Broadly speaking, causality is the relation between a cause and its effect. Of the non-hierarchical relations in EcoLexicon, causality is one of the most important. Obviously, the environment is conceived as a process where causes and effects are at the core of any event. Not surprisingly, causal relations are also crucial for other difficult tasks in NLP, such as question answering (Girju, 2003).

The extraction and representation of causality have been studied from a wide range of disciplines and perspectives. These include: (i) Cognitive Linguistics, as reflected in Talmy's Force dynamics (2000), (ii) Artificial Intelligence, in different NLP applications; (iii) Philosophy and Psychology (White, 1990), etc. All these studies affirm that there are many ways to express causation since it can be expressed in passive, active, subject-object, nominal or verbal propositions. Moreover, causes and effects have very diverse syntactic representations. More specifically, causation is not only expressed by constructions such as *due to* or *because of*, but also by causative nouns (*cause* or *consequence*) and verbs. Although there are many causative verbs (e.g. *cause, generate*, *lead*, *produce*, etc.), their syntactic behavior can vary. As a result, one single grammar is not sufficient to formalize their complementation structures.

This has led researchers to classify causal relations in different facets. For example, Blanco et al. (2008) classified these relations in *influence*, *condition*, *consequence* and *reason*. In contrast, the classification in Nastase (2003) is based on *cause*, *effect*, *purpose*, *entailment*, *enablement*, *detraction* and *prevention*. For Khoo et al. (2002), causation is also complex and multifaceted. They use templates for each causal category involved in the relation (*cause*, *effect*, *subjects involved*, *condition*, *modality*) and provide a classification of explicit patterns, such as adverbial (*so*, *hence*, *therefore*), prepositional (*because of*) and subordination (*as*, *since*) causal links, clause integrated links (*that's why*, *the result was*), causative verbs (*break*, *kill*), resultative constructions, conditionals and causative adverbs, adjectives, and prepositions.

Girju (2003) also states that causative constructions may be explicit or implicit. Her work focuses on explicit but ambiguous verbal causation patterns. She provides a list of 60 causative verbs and classifies them into simple causatives (*cause*, *lead to*, *bring about*, *generate*, *make*, *force*, *allow*); resultative causatives (*kill*, *melt*, *dry*, etc.) and instrumental causatives (*poison*, *hang*, *punch*, *clean*) This identification of causes and effects is derived from the transitivity of WordNet verbs.

3.1 Causal grammars for EcoLexicon

In EcoLexicon, we have developed a series of KP-based micro-grammars with the help of NooJ, a development environment used to construct large-coverage descriptions of natural languages and apply them to large corpora (Silberztein, 2003). The main advantages of NooJ grammars over manual searches based on regular expressions are recursivity as well as the possibility of annotating the corpus with different tags that can be reused in batch processing tasks. We used NooJ parser to identify causal syntactic structures in a 1,200,000 word corpus. The corpus was manually classified into four contextual domains, each of approximately 300,000 words: ATMOSPHERIC SCIENCES, COASTAL ENGINEERING, OCEANOGRAPHY, and SOIL SCIENCES.

As previously mentioned, causation can be expressed in many different ways. Moreover, the semantic roles and features of the elements in a causal proposition, as well as their syntactic behaviour, can change, depending on the structure and order. For instance, in the proposition <X *causes* Y>, X is the CAUSE and Y the EFFECT, whereas in <X *is caused by* Y>, X is the EFFECT and Y is the CAUSE. This is why we have developed an array of micro-grammars for the causal realizations rather than only one. Apart from searching for the causal KP, we also wanted to

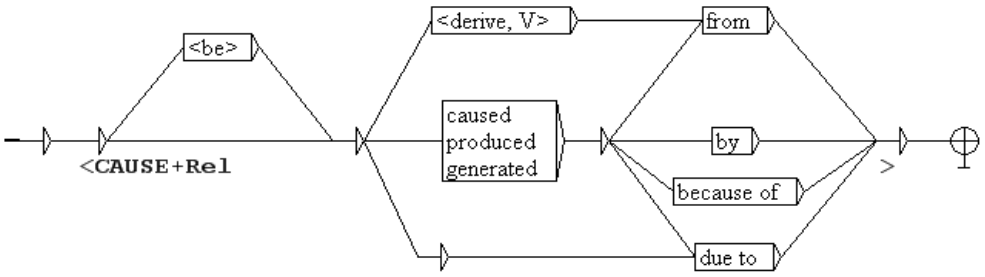


Figure 4: Core grammar of the causal relation

Flash flooding due to extremely heavy rains
Most storm-related damage was caused by wind, wind-blown rain and tornadoes
Sustained winds of tropical storm force produced by Rita
The most familiar sea level changes are produced by astronomical tides
Earthquakes are shock waves caused by abrupt movements of the earth's crust
Local wind patterns (sometimes caused by structures and urban development)
Sediment fluxes generated by incident waves
Currents are usually due to tides| and river flows
Internal waves are generated by wind energy
Tsunami can also be caused by landslides
Vapor transfer in soil due to air movement

Figure 5: Causal propositions matching <CAUSE+Rel> grammar

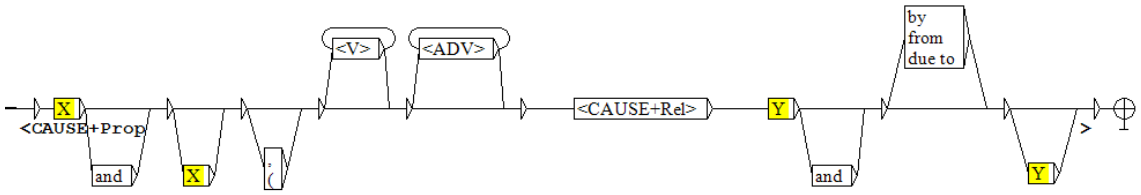


Figure 6: Grammar for causal propositions

extract the elements involved, whether they are causes or effects and regardless of whether they are already stored in our TKB or not.

Thus, when the corpus is matched with the graph-based micro-grammar structures, it is possible to annotate the corpus and extract the entire causal proposition as well as the environmental terms acting as causes and effects.

So far, we have developed five micro-grammars for the following constructions: <X *causes* Y>, <X *caused by* Y>, <X *is the cause of* Y>, <*the cause of* X *is* Y>, and <X *causes* Y *to* Z>. Of course, they are not limited to the verb or noun cause, but also include other causative verbs and nouns. However, we did not include all 60 verbs found in Girju (2003) because each requires a different treatment and will be dealt with separately in the future. Moreover, some of these verbs correspond to other domain-specific relations in EcoLexicon.

This first approach to causation only focuses on the construction <X *caused by* Y>. Despite the many other ways to approach causation in the corpus, this pilot study yielded surprisingly rich results.

For efficiency reasons, the first step was to elaborate a grammar that formalized the most basic sense of causation (Figure 4). This grammar extracts causal links by following different paths. As shown in Figure 4, causation can be expressed by: the participle of *cause*, *produce* and *generate* (optionally preceded by *to be* in any of its inflected forms), and followed by one of the four prepositional constructions. However, it can also be expressed by *derive*, in any of its inflected forms followed by the preposition *from*, or by the adjectival phrase *due to*. We located all of the occurrences matching this grammar and annotated them with the tag <CAUSE+Rel>. From the entire corpus, we extracted 960 causal occurrences, and thus found meaningful causal sentences such as those in Figure 5.

However, not all of them were found to be valid causal propositions, since sometimes the causal expression did not link two specialized terms, such as those cases where x is expressed as *this*, *that*, etc. Thus, we designed a more complex micro-grammar that reused the annotation <CAUSE+Rel> as the link between X (EFFECT) and Y (CAUSE) (Figure 6).

This grammar contemplates the possibility of having more than one effect and/or cause in the same causal proposition (i.e. ***chemical solution*** *and* ***mechanical abrasion*** *caused by some organisms* or *dune erosion produced by* ***storm waves*** *and* ***water level***). This is why X

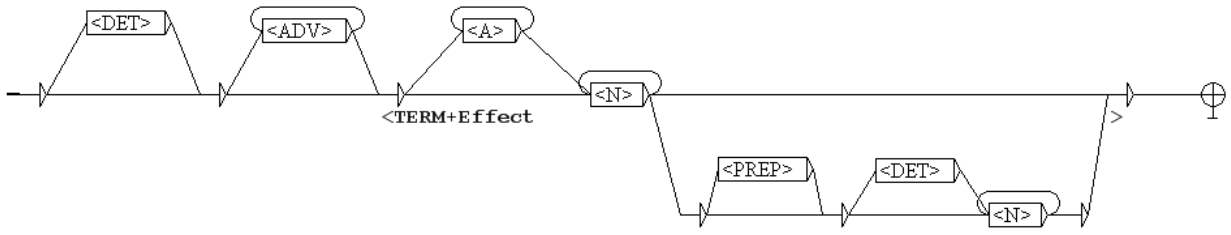


Figure 7: Grammar for <TERM+Effect>

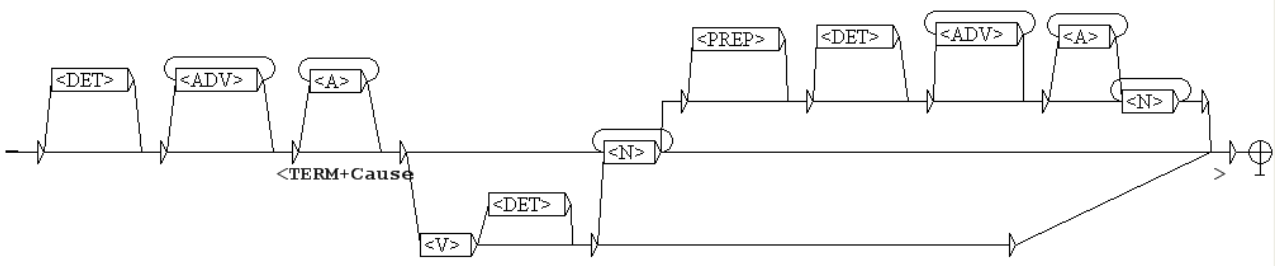


Figure 8: Grammar for <TERM+Cause>

and Y appear twice joined by the conjunction *and*, along with certain prepositions already used in <CAUSE+Rel>. It also includes punctuation marks, such as a comma and a bracket, since they often appear between effects and causal links, as in *local wind patterns (sometimes caused by structures and urban development)*. Moreover, it also accounts for the occurrence of one or more verbs (<V>*) and/or one or more adverbs (<ADV>*) between the effects and <CAUSE+Rel>.

As a result, this grammar is able to identify sentences like *continental glaciers **possibly** caused by a warming climate*, *coastal erosion **may be mainly** produced by wave attack*, or *tsunami **can also be** caused by landslides*. Note that in *can also be*, can also corresponds to <V><ADV> and *be* is matched through the <CAUSE+Rel> grammar. Once identified, they are annotated as <CAUSE+Prop>.

The elements highlighted are two different sub-graphs describing the possible syntactic structure of both X (EFFECT) and Y (CAUSE) as specialized terms (Figures 7 and 8).

As is well known, specialized knowledge units are very often multi-word terms composed of two nouns (*beach erosion*), a combination of adjectives and nouns (*detached breakwater*) or prepositional sentences (*the gravitation of the moon*). Moreover, when they are inserted in a text, they can also be modified by adverbs or adjectives that, strictly speaking, are not part of the terminological phraseme. This is why they are not included in the annotations <TERM+Effect> and <TERM+Cause>, but do appear in the grammar in order to identify the whole proposition.

These structures are capable of identifying various causes and effects as multi-word terms. In ***delta land loss** caused by **rising sea level***, the effect is identified by following the path <N>* and the cause through <A><N>*. In ***cliff retreat**, caused by unusually **severe winter storms***, the effect and the cause are recovered through the paths <N>* and <A><N>*, respectively. This is possible despite the presence of an adverb (*unusually*) that matches the grammar but is not recovered as part of the term. More complex sentences can also be found, such as ***rates of subsidence** caused by **compaction of newly deposited sediment***, where the effect now follows the path <N><PREP><N> and the cause <N><PREP><A>*<N>. Furthermore, causes are defined by means of an additional path that includes a verbal proposition in order to identify phrases like ***environmental damages** caused by **dredging the river*** (<V><DET><N>).

3.2 Causal propositions in EcoLexicon

A search for all <CAUSE+Prop> annotated sentences gave 347 propositions, which were filtered out from the initial 960 occurrences through the formal description of effects and causes as specialized terms (<TERM+Effect> and <TERM+Cause>).

These three tags thus allow the extraction of all meaningful causal propositions for each concept in the corpus and automatically display them to users. Even more interestingly, it is also possible to extract all effect-cause pairs, as well as to measure the prototypicality of certain causal propositions, in each domain.

For instance, Table 1 gives a simplified classification of the most common causes and effects of all four contextual domains.

	CAUSE	EFFECT
ATMOSPHERIC SCIENCES	Tropical cyclones, swells, hurricane, wind, storm, storm surge, heavy rains, floods, typhoon, thunderstorms	Floods, storm surge, waves, tropical storm force winds, rise in ocean level, swells, adiabatic changes
COSTAL ENGINEERING	Glaciers, tides, gravitation, tropical storms, wind, groundwater withdrawal, tectonic movements, dams, rising sea level, changes in wave energy, tidal currents, offshore transport, recession of the beach, seawall, waves, scour, wave action, wave attack, longshore transport, erosion	Fall of water levels, wind, water level changes, eustatic rise in sea level, tsunamis, salt weathering, ocean waves, changes in sea level , antidunes, waves, currents, longshore sand transport, erosion
OCEANOGRAPHY	Tectonic forces, seawater, wind energy, wind, landslides, tidal currents, gravitation, wave swell, faulting	Storm surge, tsunami, waves, tides, wind, estuaries
SOIL SCIENCES	Electrical polarity of the water molecule, vegetation canopy, pressure gradient, gravitation, downward seepage, vapor pressure, osmosis, wind	Rise of the water table, sand columns, intermolecultar forces in liquid water, transpiration, wind

Table 1: CAUSES and EFFECTS in four contextual domains

As can be observed in Table 1, the four domains share many of the same causes and effects detected by the <X *caused by* Y> proposition. Moreover, the multidimensionality of the environmental domain is reflected in certain concepts that can act both as cause and effect even within the same domain (WIND, TIDE, CURRENT, FLOOD, etc.). Interestingly enough, WIND can be cause and effect in all four domains. However, its prototypical role changes across them. Figure 9 and 10 show the standard score of WIND as an effect and as a cause in each of the corpora. The standard score, retrieved thanks to NooJ's statistical module, shows the standard deviations of the occurrences that are above or below the mean. This is similar to the concept of prototypicality used to recontextualize semantic networks in EcoLexicon. Thus, based on Figures 9 and 10, the propositions in which WIND is an effect mostly appear in ATMOSPHERIC SCIENCES texts, whereas those in which WIND is a cause primarily occur in ATMOSPHERIC SCIENCES and OCEANOGRAPHY texts. Therefore, the concept is recontextualized in semantic networks accordingly.

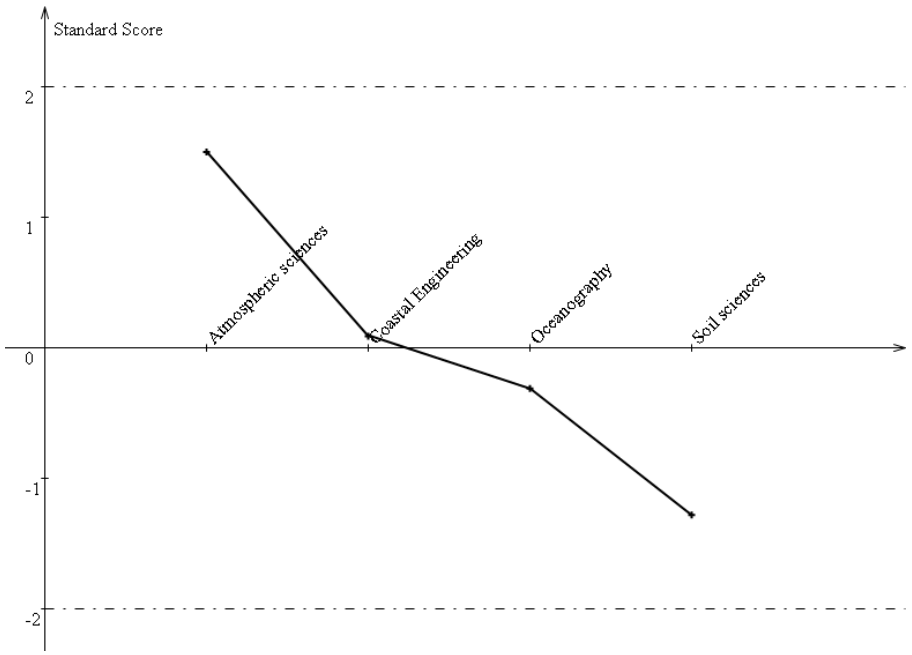


Figure 9: Prototypicality of WIND as an effect

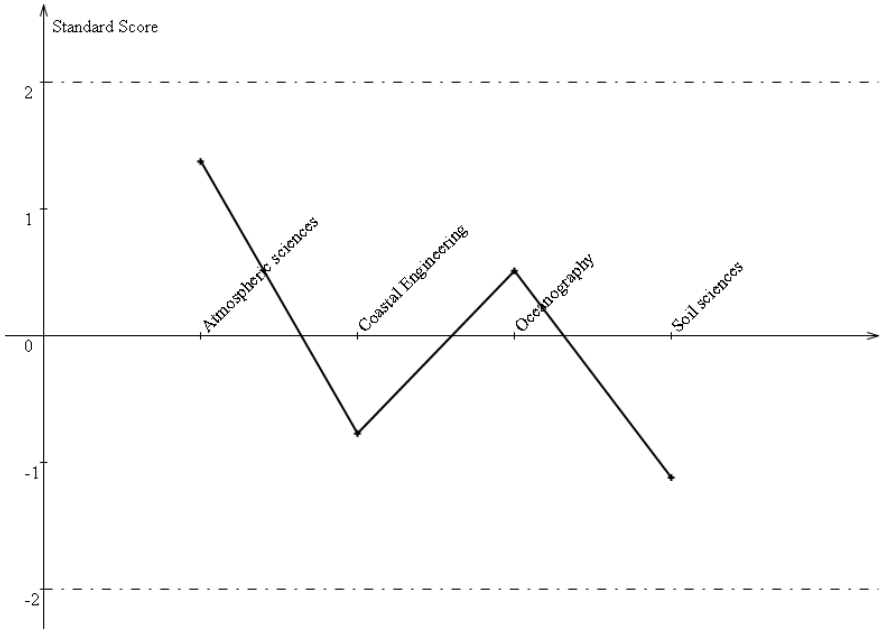


Figure 10: Prototypicality of WIND as a cause

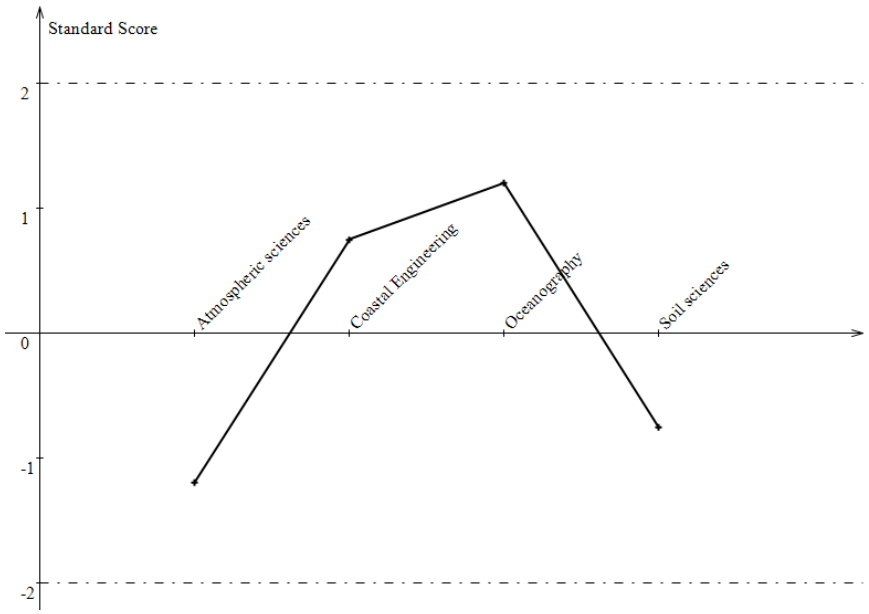


Figure 11: Prototypicality of <STORM SURGE *caused by* WIND>

However, this does not mean that each causal proposition in which WIND is a cause only occurs in ATMOSPHERIC SCIENCES and OCEANOGRAHY. Regarding the concrete WIND-related proposition <STORM SURGE *caused by* WIND>, the results show that it should not only be included in the recontextualized semantic network of OCEANOGRAPHY (and not in that of ATMOSPHERIC SCIENCES), but also in that of COASTAL ENGINEERING. This is why contextual constraints are not applied to individual concepts nor to semantic relations, but to complete and concrete conceptual propositions.

4. Conclusion and future work

In this paper we have shown how KP-based corpus analysis can be enhanced through the formal description of the syntactic structures of KPs and the help of NLP applications. Although manual work is still necessary to discover new patterns that reflect semantic relations in real texts, the knowledge thus acquired can be reused in automatic procedures. Otherwise, knowledge representation in lexical resources would be overly dependent on intuition.

In the near future, these patterns will be applied to the whole corpus in EcoLexicon. Once the corpus is classified in contextual domains, it will be proccessed using these causal micro-grammars, and new ones will be designed for other semantic relations in our TKB. This is a cyclic process since the application of relational micro-grammars to the most prototypical term pairs in each domain will also validate the categorization of the corpus.

A further step will be to identify possible cases of noise and silence and finally measure the precision and recall of the results with a gold standard. The disambiguation of polysemic structures also remains a challenge. Apart from polysemic KPs, specialized terms may also yield confusing results. For instance, when searching for the prototypicality of WAVE-related propositions, the SOIL SCIENCES domain shows false positives. The reason for this is that *wave* is a very common term in this domain, but only in its physics sense and not in its sea-related sense. Our intuition is that these problems could be solved by adding a semantic component to the grammars. As Girju and Moldovan (2002) state, semantic features are essential to constrain which entities will be efficiently linked through causation. Although these authors use a set of features from WordNet for this purpose, we plan to implement a NooJ-based dictionary containing all of the terms in EcoLexicon as well as the semantic features that define our concepts and categories.

5. Acknowledgements

This research has been carried out within the framework of the project RECORD: Representación del Conocimiento en Redes Dinámicas [Knowledge Representation in Dynamic Networks, FFI2011-22397], funded by the Spanish Ministry for Science and Innovation.

6. References

Aussenac-Gilles, N.; Séguela, P. (2000). Les relations sémantiques : du linguistique au formel. *Cahiers de Grammaire* 25, *Sémantique et Corpus*, 175-198.

Barrière, C.; Agbago, A. (2006). TerminoWeb: a software environment for term study in rich contexts, *International Conference on Terminology, Standardization and Technology Transfer,* Beijing, 103-113.

Barrière, C. (2004). Building a concept hierarchy from corpus analysis. *Terminology* 10: 2, 241-263.

Blanco, E.; Castell, N.; Moldovan, D. (2008). Causal Relation Extraction. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco.

Bowker, L. (2004). Lexical knowledge Patterns, Semantic Relations, and Language Varieties: Exploring the Possibilities for Refining Information Retrieval in an International Context. *Cataloging and Classification Quarterly*, 37(1): 153 - 171

Cimiano, P., Staab, S. (2005) Learning Concept Hierarchies from Text with a Guided Agglomerative Clustering Algorithm, *Workshop on Learning and Extending Lexical Ontologies with Machine Learning Methods*, Bonn.

Condamines, A. (2002). Corpus analysis and conceptual relation patterns. *Terminology* 8:1, 141-162.

Embarek, M.; Ferret, O. (2008). Learning Patterns for Building Resources about Semantic Relations in the Medical Domain. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 08).* Marrakech, Morocco: ELRA.

Girju, R. (2003). Automatic Detection of Causal Relations for Question Answering. In the proceedings of the *41st Annual Meeting of the Association for Computational Linguistics (ACL 2003), Workshop on "Multilingual Summarization and Question Answering - Machine Learning and Beyond"*.

Girju, R.; Moldovan, D. (2002). Text Mining for Causal Relations. In *Proceedings of the International Florida Artificial Intelligence Research Society (FLAIRS 2002)*, Pensacola, Florida.

Hearst, M.A. (1992). Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-1992)*.

Khoo, C.; Chan, S.; Niu, Y.; Ang, A. (1999). A method for extracting causal knowledge from textual databases. *Singapore Journal of Library and Information Management*, 28, 48-63.

Khoo, C.; Chan, S.; Niu, Y. (2002). The many facets of the cause-effect relation. In R. Green, C.A. Bean and S.H. Myaeng (eds.), *The semantics of relationships: an interdisciplinary perspective*. 51-70.

León Araúz P.; Faber, P. (2010). Natural and contextual constraints for domain-specific relations. In *Proceedings of the Workshop Semantic Relations. Theory and Applications*, ed. Verginica Barbu Mititelu, Viktor Pekar, and Eduard Barbu, 12-17. Valletta.

León Araúz, P; Reimerink, A. (2010). Knowledge extraction and multidimensionality in the environmental domain. In *Proceedings of the Terminology and Knowledge Engineering (TKE) Conference 2010*. Dublin: Dublin City University.

León Araúz, P.; San Martín, A. (in press). Distinguishing polysemy from contextual variation in terminographic definitions. In *Proceedings of the 10th International Conference of AELFE*, Valencia: AELFE.

Marshman, E.; Morgan, T.; Meyer I. (2002). French patterns for expressing concept relations. *Terminology* 8:1, 1-29.

Marshman, E. (2002). The cause relation in Biopharmaceutical Texts: Some English Knowledge Patterns. In *Proceedings of Terminology and Knowledge Engineering, TKE 2002*, 89-94.

Meyer, I. (2001). Extracting knowledge-rich contexts for terminography: A conceptual and methodological framework. In Bourigault, D., Jacquemin, C. and L'Homme, M.C. (eds.), *Recent Advances in Computational Terminology*. Amsterdam: John Benjamins. 279-302.

Nastase, V.(2003). *Semantic Relations Across Syntactic Levels*. Ph.D. Dissertation, University of Ottawa.

Rosario, B. and Hearst, M. (2004). Classifying Semantic Relations in Bioscience Text. In *Proceedings of ACL 2004*.

Silberztein, M. (2003). *NooJ Manual*. Available at: http://www.nooj4nlp.net/NooJManual.pdf

Talmy, L. (2000). *Toward a cognitive semantics*. MIT Press

Vintar, S.; Buitelaar, P. (2003). Semantic relations in concept-based cross-language medical information retrieval. In *ECML/PKDD Workshop on Adaptive Text Extraction and Mining (ATEM)*, Germany.

White, P. (1990). Ideas about causation in philosophy and psychology. *Psychological Bulletin*, 108 (1): 3-18

A Pilot Study: Deriving a User's Goal Framework from a Corpus of Interviews and Diaries

Scott Piao[1], Diana Bental[2], Jon Whittle[1], Ruth Aylett[2], Stephann Makri[3], Xu Sun[4]
[1]School of Computing and Communications
Lancaster University
Lancaster, UK
[2]School of Maths and Computer Science
Heriot-Watt University,
Edinburgh, UK
[3]UCL Interaction Centre
University College London,
London, UK
[4]Human Factors Research Group, Department of Mechanical, Materials and Manufacturing Engineering,
University of Nottingham,
Nottingham, UK
[1]{s.piao,j.n.whittle}@lancaster.ac.uk, [2]{r.s.aylett,d.s.bental}@hw.ac.uk, [3]s.makri@ucl.ac.uk,
[4]xu.sun@nottingham.edu.cn

Abstract

This paper describes pilot work in which we explore the feasibility of deriving a goal framework for the potential users of applications employing a grounded theory method based on a corpus of empirical data. The issue of developing and applying human goal frameworks has been studied in a number of areas, such as artificial intelligence and information seeking. But most existing goal frameworks are either constrained to a few information search related goals or mainly reflect highly abstract psychological motivations, and hence are not readily applicable to the applications which need to deal with complex practical users' goals. In this study, we employ corpus-based approach for goal framework development, and identify goal concepts and analyse semantic relations among them based on a collection of interview and diary transcripts. We suggest that our approach provides a feasible way of deriving goal frameworks for practical purposes as the corpus data tend to closely reflect the users' concrete requirements. Furthermore, our study reveals the need for more corpus resources for human goal analysis and automatic detection.

1. Introduction

It is an important issue to identify and compile human goal frameworks for intelligent systems, and it has been studied in a number of research areas such as psychology, artificial intelligence and information seeking (Chulef et al., 2001; Mueller, 1990; Amin et al., 2008). While the earlier work introduced various frameworks of human goals, we find it difficult to apply them for practical purposes.

A goal framework is a conceptual model in which human goal concepts are categorised and organised in a certain structure, often in hierarchical structure. A typical example is the goal taxonomy available from the PsychWiki website (http://www.psychwiki.com/wiki/Goal_Taxonomy), in which human goals are extracted and organised from a psychological point of view in a three-layered taxonomy. For example, it consists of three top goal categories of "TO BE HAPPY", "TO FEEL MORAL" and "OLDER CATEGORIES", which are further divided into sub-categories such as "to achieve something", "to help others", "to feel autonomy/self direction" etc. Not all goal frameworks are as complex as this one. Some of them are targeted at very specific domains, tasks or contexts, and consist of a small number of goal categories needed to address practical needs, as will be explained later in this paper.

The goal framework we seek to obtain or develop is related to SerenA Project (http://www.serena.ac.uk) in which we explore and develop methods and application software for automatically recommending potentially serendipitous connections of information sources and people. The potential goals pertinent to the users of the software (termed *users' goals* hereafter) are one of several dimensions, such as users' interests and preferences, along which we search for such connections. For example, if we can identify a user's goal, such as attending a conference in near future or planning to buy a new car, we may be able to find and recommend information or people that may be of interest to this user. In this work, our focus is on how we can find such users' goals and provide a goal framework for such practical applications.

It should be noted that, although our work was initially started to address the requirements of our current project, the goal information can have a wide range of applications in many other information systems such as social networking applications. For example, the social networking site "43 Things" (http://www.43things.com) connects users based on their goals, which need to be typed in as plain natural language text by the users. An appropriate goal framework would bring benefits to such

systems.

It is usually preferable to re-use existing frameworks rather than creating a new one for each application. The difficulty we face in re-using the existing goal frameworks stems from two aspects. Firstly, some of them are hard-coded in the logic rules for dealing with very specific pre-defined target users or domains (Mueller, 1990) and hence it is difficult to port them for new domains and tasks. Secondly, some of them reflect highly abstract levels of human psychological motivations, such as the PsychWiki Goal Taxonomy, and it is difficult to map these abstract motivational categories to users' more concrete goals.

So an interesting issue arises here: Is it possible to derive a framework of users' goals for practical purposes via an empirical approach, such as deriving it from a corpus of empirical data? The critical issue here is that the user goals need to be at a fairly concrete level, rather than a highly abstract level, to cater for needs of practical applications. For example, we need to identify concrete goals such as "travelling to a place" or "attending a conference" rather than abstract ones such as "to feel loyal" or "to be stimulated". Given the nearly boundless scale and complexity of such concrete goals, it would be impractical, if not impossible, to exhaustively list them. We propose that a practical solution to this issue is to derive limited goal frameworks from a corpus of empirical data, which meets the requirement of practical applications for constrained domains and contexts. For example, in our study we used a collection of interview and diary transcripts of some university students and researchers as the corpus, which contains information about their goals. As the interviewees represent a target user group of the tools under development in our project, we assumed it is possible to identify users goals, at least part of them, by analysing the data (see Section 3 for details of the data).

Other issues involved in our work include a) how goals are expressed in text; b) How to organise and structure the identified goal concepts based on semantic relations among them; c) how to keep a balance between making the goals concrete enough to allow useful inferences and being abstract enough to support generalisable inferences based on the goals.

As far as we know, there is no published work addressing these issues. In our pilot study to be presented below, we explore the above issues mainly based on interview and diary transcripts as the corpus of empirical data. Our work shows that our approach can provide a practical solution to the issue of providing goal frameworks for applications for which re-usable frameworks do not exist.

2. Related work

Over the past years, there has been an increasing awareness of the user's goals and intentions and such information has been proven important in a variety of applications which support information search, retrieval (Rose and Levinson, 2004; Strohmeier, 2008; Strohmeier and Kröll, 2012) and social networking (*43Things* Website mentioned earlier).

The users' goals and intentions can help determine what information is relevant them, but it is not always straightforward to determine their intentions or to identify the information that best matches those intentions. For example, more often than not, users do not express their intentions explicitly in web queries, and web pages are typically tagged with descriptions of their content without specifying the purposes to which their content may usefully be put (Strohmeier et al., 2008).

Various attempts have been made to bridge this gap. For instance, GOOSE (Liu et al., 2006) is a search tool that allows users to express different types of goals as part of their query and applies templates to expand the query appropriately to match sites more accurately. Strohmeier (2008) takes a social tagging approach which provides a mechanism that encourages users to add "purpose tags" to sites in addition to the usual content tags, allowing the search tool to extend queries with purpose information. Faaborg and Lieberman's (2006) goal-oriented web browser takes a 'programming by example' approach to gathering and inferring a user's goals. Depending on the identified user's goal, a retrieved page may offer links to different types of information.

Furthermore, there have also been various attempts to classify the goals of users seeking and consuming information on the web. Rose and Levinson (2006) and Broder (2002) broadly distinguish three types of web search: navigational (with the intention to access a specific website, often the homepage of an organisation); informational (finding information about a topic or an item, such as locating a product or service); and resource (where the resource itself may be online, such as playable music). Kellar et al. (2007) offer a similar classification scheme of information-seeking behaviours on the web, containing four main categories: information seeking, browsing, information exchange and maintenance. GOOSE mentioned earlier supports five common types of search goals, without claiming that these types are exhaustive: (i) I want help solving a problem; (ii) I want to research…; (iii) I want to find websites about…; (iv) I want to find other people who…; (v) I want details about a product/service. Nevertheless, not all of the applications mentioned above support users' goals and intentions or represent goals explicitly. For example, Faaborg and Liebermann's 'programming by example' approach does not attempt to classify goals explicitly, but assumes that similar user intentions can be applied to semantically similar items.

Other similar efforts include developing comprehensive goal frameworks in the form of taxonomies. For example, Chulef et al. (2001) developed a hierarchical taxonomy of 135 human goals, which are grouped based on similarity judgments. Various factors were considered in structuring

the goals, such as gender and age. The PsychWiki taxonomy provides another similar goal framework. As mentioned previously, these taxonomies reflect rather abstract psychological concepts of goals.

The previous work mentioned above address the issue of development and application of goal frameworks from various angles. However, they do not meet the requirements of our application, being either too domain-specific or too abstract.

Aiming to detect and recommend serendipitous connections of people as well as information sources, we need to identify rather concrete goals of users, which we found are not covered by any of the existing goal frameworks. In our work, we adopt an approach different from earlier work mentioned above in that we attempt to derive users' goals by observing and analysing empirical data collected from the potential users concerned. By doing so, we aim to investigate the issue of developing practically useful and applicable goal frameworks for individual applications based on corpus analysis, for which no existing frameworks are applicable.

3. Identifying users' goals based on empirical data

The method we followed for developing users' goal framework based on corpus data is as follows:

1) Gather a corpus of empirical data from relevant sources, such as requirement documents, user interviews and diaries (Sun et al., 2011; Makri and Blandford, 2012), which contain information about goals of the users of the application software under development.
2) Identify syntactic units (mostly sentences) expressing goals in the corpus and assign them with goal categories. This provides a basis for compiling a goal framework.
3) Group and organise the identified goal categories into a framework (a taxonomy in this particular case) based on semantic relations, in this particular case a taxonomy.

The following sections describe the process in details.

3.1 Data for goal analysis

With regards to the data gathering, we used transcripts of a set of audio diaries and interviews produced in our project, in which interviewees are asked to talk about their serendipity experiences. These interviewees were conducted as part of 2 separate studies of research students and academic researchers. Both of these studies were aimed at capturing their experiences of serendipity. During the first study (see Sun et al., 2011), 11 participants used a mobile diary application to record their experiences of serendipity over the period of a week, and were subsequently interviewed about these experiences. During the second study (Makri & Blandford, 2012), interviews were undertaken with 23 researchers in 11 disciplines, during which the researchers were asked to discuss memorable experiences of serendipity. The interviewees were not directly asked to describe their goals. Instead, their goals became apparent through the examples of serendipity that they provided. Through their provision of these examples, goals that were achieved or supported by the interviewees' serendipitous experiences emerged. So did other goals that they were pursuing when serendipity struck. As the interviewees represent potential users of the tools under development in our project, this makes the interview data suitable for reflecting practical information-oriented user goals.

Another reason for selecting this data is for its informal nature. As transcripts of spoken language, the data contain grammatical "noise" and non-standard expressions, e.g.

> *"Okay, so neither do I but yes, that's one big dilemma I have when I will be talking to my design team because they need to distil something about what people understand about serendipity."*

While such a feature of the data causes difficulty for analysis and would be normally considered as problematic for goal-extraction purposes, it can actually provide potential benefit in terms of related tool development. As the data can be used for training tools for automatic goal detection, its informal style will allow us to develop tools which can potentially cope with similarly "noisy" mediums such as social media (tool development is beyond the scope of this paper).

3.2 Manual analysis of data

We preformed an analysis of the 11 diaries and corresponding interviews from the diary study, and five of the interviews from the second study. The raw interview data was in the form of dialogues, in which interviewer asks some questions and the interviewee provides detailed response and explanations. As mentioned, the theme of the interviews is serendipity, reflecting the main research theme of our project. Therefore, we expected to find various goals in relation to serendipity in the data, and we used the interview transcripts as a corpus for deriving a goal framework that is applicable to the application domain represented by the data.

We found that the goals are conveyed by different syntactic units, including clauses and sentences. In some cases more than one sentence is involved in expressing a goal. For the convenience of analysis, we used the sentence as the main unit for analysis. Therefore, the goal information is mostly annotated for sentences. In some cases, a sentence can be very long, which mostly are juxtaposed sentences with sentence termination punctuations missing due to transcription errors. In such cases, we selected clause/s which are closely relevant to a given goal. In exceptional cases where more than one sentence is closely related to a given goal, we select them as the annotation unit.

In terms of goal categories, we followed a Grounded Theory approach (see Corbin & Strauss, 2008), more specifically an emergent qualitative coding approach. That is, we did not start the analysis with any pre-defined user goal framework. In fact, as we explained earlier, there is no such re-usable framework available. Our approach was to create goal labels/tags when we came across new goals mentioned in the data. For example, we used the label "FIND STH" to annotate those sentences that convey the goal of finding something, as is the case for the sentence "I am looking for module information from different sources". We kept the goal semantic categories at very concrete level, but abstract enough to cover synonymous linguistic expressions. For example, the category of CONTACT ENTITY is used to group expressions such as "contact … ", "get in touch with …", "email someone …" etc. As the analysis proceeded, we obtained a set of goal categories/tags, which covers a range of goals found in the data and provide a basis for developing a goal framework.

The main reason for adopting the Grounded Theory approach is the lack of a reusable goal framework and the complexity of potential human goals. As explained earlier, we do not seek to develop an all-round, complete human goal framework at highly abstract level. What we desire is a set of "low-level", fairly concrete goal categories such as finding something or attending a meeting etc. As there can be huge number of such goals, it would be nearly impossible to enumerate them. Consequently, it is impractical trying to pre-define a comprehensive goal framework covering all foreseeable needs and contexts. Therefore, we suggest that a more practical approach is to build up a goal framework from bottom based on what can be observed and identified in a corpus of empirical data, i.e. data containing information about goals that the users of the tools might come across, the diary and interview data in our particular case.

Two researchers dedicated 3 weeks to manual analysis and annotation of the data, producing 1,155 annotated text units (mostly sentences). This shows that the annotation process can be conducted relatively quickly. Note that not every sentence mentions goals, rather, such sentences are scattered thinly across the interview data. Hence the researchers had to read through every sentence in search of them. For a larger scale of such annotation, substantial amount of effort would be needed. The annotation phase is intended to derive an initial structure of goal framework which can then be supplemented by automated analysis.

As the annotation was carried out by 2 researchers individually without a pre-defined goal category framework, some inconsistency of annotation occurred during the analysis process. For example, different labels/tags were used for the same goals, or the same tags were used differently. We carried out frequent cross-checking to resolve these inconsistencies.

As a result, from the annotated sentences, we collected a total of 169 goal categories. After a frequency analysis, we found that 68 categories occur at least 3 times in the data. As we intend to focus on those goals that are more likely to appear in practical situations, currently we mainly consider those categories of frequencies above 2, i.e. 68 of them are considered for the initial prototype goal framework. Table 1 lists some top-frequent goal categories, in which the first column shows frequencies.

Freq	Goal Category
98	FIND STH
61	PLAN TO DO STH
59	STUDY STH
53	CONNECT ENTITIES
49	NOTE STH
38	CONSIDER STH
32	TRY/ATTEMPT TO DO STH
30	INTEND TO DO STH
28	READ STH
25	TALK TO PEOPLE
24	RECOMMEND STH
22	INVESTIGATE STH
21	MEET PEOPLE
20	USE STH
19	WANT STH [Goal Cue]
18	FILTER STH
18	SOLVE STH
17	ENCOURAGE STH
16	DISSEMINATE STH
15	BE QUALIFIED IN STH
14	DEVELOP STH
14	OBTAIN STH
14	WANT TO DO STH
13	GO TO PLACE
12	LOOK AT STH
11	ATTEND STH
11	SUGGEST STH
9	CONTACT ENTITY
9	CREATE STH
9	PURCHASE STH
8	TELL SOMEONE ABOUT STH
7	ENGAGE IN STH
7	LISTEN TO STH
7	MENTION STH
7	SHARE STH

Table 1: Goal categories which have frequencies greater than six.

Due to the limited size of data available for the analysis, the resultant goal categories are by no means representative of the user goals. Nonetheless, the highly frequent goal categories, such as "FIND STH" (f=98), "PLAN TO DO STH" (f=61), "STUDY STH" (f=59), "NOTE STH" (f=49) etc. definitely reflect some primary user goals expected of university students and researchers, from whom the interview data were collected. In fact, many of the frequent goal categories are also applicable to general users and general contexts, and can be ported to other application domains such as social networking.

One may notice that the goal category labels mainly specify predicates, such as FIND, MENTION etc, without specifying their objects such as STH at this stage. This is mainly because of the uncontrollable diversity of the object types. Is all of them are to be specified, it would cause difficulty in categorizing the goals with finite range of spectrum. Therefore, we leave the objects, as well as the subjects (by default the users) of goal categories, as slots to be filled by separate process, which would entail detailed semantic analysis of the text (beyond the scope of this paper).

3.3 Structuring the goal categories into a goal taxonomy

The goal categories collected from the interview corpus data, as well as some additional ones suggested by application domain experts, are organised into a goal taxonomy, which will provide a framework for further annotation and classification of new text. While there can be numerous different criteria for structuring the goal categories, currently we group and organise them mainly based on semantic hyponymous relations into a crude hierarchically structured taxonomy that reflects the application domain.

First of all, we identified four categories which mainly function as indicators of goals. I.e. they themselves may not be goals, but they indicate that what follows is likely to be a goal. For example, the sentence "I'd like to buy an iPhone next month" implies both INTEND TO DO STH and PURCHASE STH. But the former is not a concrete goal, rather it mainly implies that the following action "to buy …", or PURCHASE STH, is a goal. We define such categories as *Goal Indicators*, as shown below:

- PLAN TO DO STH
- TRY/ATTEMPT TO DO STH
- INTEND TO DO STH
- WANT TO DO STH

The remaining categories other than the Goal Indicators are actual concrete goals. We divide the concrete goal categories into *General Goals* and *Domain Specific Goals*. Here the General Goals refer to the goals that can occur in general contexts in daily life such as "going somewhere" (GO TO PLACE) or "buying some food" (PURCHASE STH). On the other hand, the Domain Specific Goals mainly occur in specific contexts, such as academic research or sports. For example, "visualise data" in Design study or "win the match" in football games.

We observed that most of the goals mentioned in the corpus have a general application as well as being important within the research domain. The categories STUDY, INVESTIGATE, and DEVELOP are the most specific to research, but goals may be specific or general depending on the predicate objects and contexts. For example a researcher could FIND information that is related to their research, or they could FIND information about football and other topics of personal interest. Such a duality of many goal categories cause difficulty in organizing them in a hierarchical structure, but in the same time it can be advantageous in that the goal framework can become applicable to a wider range of application domains. A possible solution to the duality issue might be to classify such goal categories into generic or specific groups according to the type of objects and contexts of goal occurrence.

As an additional step towards a taxonomy of goals we conducted a card sorting exercise with a group of researchers, using descriptions of the goals derived from the interview transcripts described earlier. We focused on refining the groups of goal categories within research domain, which is a focus of our project. During this exercise we identified a number of groupings of goals, including:

1. Information gathering goals (such as FIND);
2. Communication and collaboration goals (such as MEET, CONTACT, RECOMMEND);
3. Producing outcomes (e.g. WRITE);
4. Analysis/synthesis (e.g. CONNECT, CONSIDER, USE).

The insight gained from the exercise helped us to further refine the structure of the goal taxonomy, particularly in grouping the research-related goals. Figure 1 illustrates the top structure of the taxonomy we propose, where the concept of THING is used as the root. Further down in the branches of generic and domain specific goals, the categories will be further clustered into sub-groups.

Appendix 1 shows a prototype goal taxonomy (subject to change and modification). In the taxonomy table, the goals are classified as domain specific goals wherever they have certain links with research activities. Many of them, in fact, can be general goals, such as MEET PEOPLE, but in order to avoid duplication, they are not included in the general goal category. By default, most domain specific goals can potentially be used as general goals.

If we compare our goal framework with the PsychWiki goal taxonomy, we can see that the PsychWiki taxonomy has little overlap with ours. The only pair of overlapping major categories are (*1.2.1.2. Communicate/collaborate*) vs. (*3.8 Communication*) in PsychWiki, with another pair of mapping minor categories of (*1.1.26: HELP PERSON*) vs. (*2.5 To help others*). This affirms our argument that the existing goal frameworks cannot cater for the needs of many practical applications.

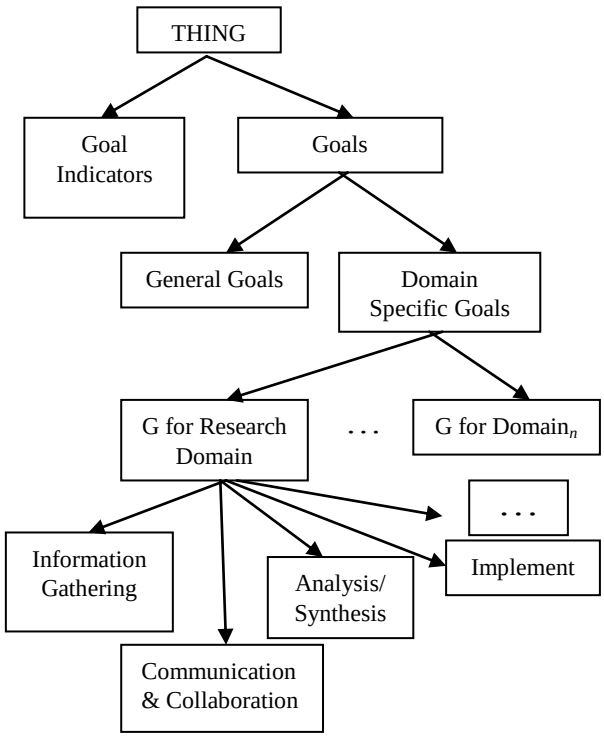


Figure 1: Outline of proposed goal taxonomy.

It should be noted that structure of the goal taxonomy we propose here is by no means the only correct one, or even our final version. There can be multiple ways of organising the goal categories, which are equally justifiable. We propose the taxonomy structure shown in Figure as a solution to our practical needs, but it will need to evolve as more goal categories become available, new semantic relations are identified among the goals, the application domain changes or need to be modified for different applications.

4. Discussion

There are a number of implications of our pilot study in terms of users' goal framework development and exploitation of corpus of empirical data for this task. Note that we do not aim to develop generic all-around human goal framework; instead, we hope to explore a practical way of compiling a goal framework that caters for the needs of specific applications for a constrained range of domains and contexts.

First of all, our experience shows that a Grounded Theory approach based on empirical corpus data can provide a practical answer to developing a goal framework for a constrained application domain. Although there can be a number of other ways of collecting the goals for similar tasks, such as asking users to explicitly create goal categories according to their needs, the corpus based approach, wherever appropriate corpus data are available, provides a reliable method for collecting core goal categories related to the given application domain.

Secondly, our approach avoids the dependency on existing or pre-defined goal frameworks. Although it would be ideal if we can re-use existing goal frameworks, our study reveals that would be difficult. Given the unpredictability of goals for different users and contexts at practical level, it would also be difficult to design a goal framework purely by theoretical reasoning. Our study shows it can be a more practical and speedy way to derive a goal framework from a corpus of empirical data, although we need to take into account the efforts needed to collect such data.

Thirdly, a benefit of our approach is that it provides an opportunity to empirically observe and study semantic relations between the goals and contexts in which a given goal occurs. Although the corpus data is devoid of real-life contexts, the surrounding narrative text provides some situational information of the goals, which is helpful in grouping and structuring the goals. Another main benefit of our approach is that it produces annotated corpus data with which tools can be developed for automatic goal detection. For many practical applications in which goal information is involved, tools will be needed for automatically identifying users' goals from natural language text generated in communications. In this regard, our approach potentially facilitates related tool training and development.

In addition, the goal framework development can benefit from the research on lexical semantic relations in corpus linguistics, as the structural relations between goals are underpinned by the semantic relations of lexicons which are used to express and describe them. Although it remains to be investigated, the hierarchical structure of the goal taxonomy, at least partially, can possibly be inferred from related lexical semantic relations.

In terms of cost efficiency, our study demonstrates that it should be feasible to develop a moderate-sized goal framework with a reasonable amount of person-hour efforts, weeks for two experienced researchers in our case. Of course, collecting the corpus data and structuring the resultant goal categories require additional efforts. Nowadays there are various techniques and tools that can assist data collection, particularly various tools for collecting audio and text messages. Such tools and techniques can assist us in collecting empirical data about users' goals with reasonable amount of efforts.

Given the pilot nature of our study and the limited size of the corpus data available, it requires further study and investigation to fully examine our approach. Nonetheless, our study supports the feasibility of our approach for the development of users' goal framework for practical applications.

5. Conclusion

In this paper, we presented our pilot study in which we explore an empirical approach to the development of a practical goal framework based on corpus of empirical

data and grounded theory.

Our research is in response to the lack of re-usable human goal frameworks for new application domains. As we have discussed, our approach can potentially bring a number of benefits for similar work in which a users' goal framework needs to be developed for an application targeting at a new user group and domain. Given the complex nature of human goals in practical scenarios, it would be difficult, if not impossible, to pre-define fit-to-all human goal framework for all foreseeable applications. Our approach can provide a practical option to address this issue.

On the other hand, our study shows that it is a non-trivial task to organise the goals into a structured framework, particularly due to the domain and context-dependent features of some goals. Although need further investigation, there is a possibility of applying the information of lexical semantic relations in structuring the goals into a framework based on goal descriptions.

As a pilot study based on limited corpus data, our findings may not be conclusive yet, and further efforts will be made to further explore our approach based on larger corpus resources and better structuring strategy of the goal categories. Furthermore, efforts will be made to develop tools for automatic goal detection based on corpus resources and goal framework.

6. Acknowledgements

We would like to thank all colleagues who provided comments and help in our study. Our study is supported by the RCUK-funded SerenA Project EP/H042741/1.

7. References

Amin, Alia, van Ossenbruggen, J., Hardman, L., van Nispen, A. (2008). Understanding Cultural Heritage Experts' Information Seeking Needs. In Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries (JCDL '08), Pittsburgh, Pennsylvania, pp. 39-47, doi:10.1145/1378889.1378897

Broder, A. (2002). A Taxonomy of Web Search. *SIGIR Forum* 36(2) ACM Press, pp. 3-10.

Corbin J. & Strauss, A. (2008). Basics of Qualitative Research (3rd Ed.): Techniques and Procedures for Developing Grounded Theory. Sage Publishers. London, UK.

Chulef, A. S., Read, S. J. and Walsh, D. A. (2001). A Hierarchical Taxonomy of Human Goals. Motivation and Emotion, 25(3).

Faaborg, A and Lieberman, H. (2006). A Goal-Oriented Web Browser. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Montreal, Canada: ACM Press, pp. 751--760.

Kellar, M., Watters, C. and Shepherd,M. (2007). A Field Study Characterizing Web-based Information-seeking Tasks. *J. Am. Soc. Inf. Sci. Technol*., 58(7).

Liu, H. and Singh, P. (2004). ConceptNet - A Practical Commonsense Reasoning Tool-Kit. BT Technology Journal, 22(7), pp 211-226.

Liu, H., Lieberman, H. and Selker, T. (2006) GOOSE: A Goal-Oriented Search Engine with Commonsense. In De Bra, P., Brusilovsky, P., and Conejo, R. (Eds.), *Adaptive Hypermedia and Adaptive Web-Based Systems* : Springer, pp. 253-263.

Makri, S. & Blandford, A. (2012). Coming Across Information Serendipitously: Part 1: A process model. To appear in *Journal of Documentation*.

Mueller, E. T. (1990). Daydreaming in Humans and Machines: A Computer Model of the Stream of Thought. Ablex Publishing Corp. Norwood, NJ, USA.

Rose, D. and Levinson, D. (2004). Understanding user goals in web search. In *Proceedings of the 13th International Conference on World Wide Web*, NY, USA: ACM Press, pp. 13-19.

Strohmaier, M. (2008). Purpose Tagging: Capturing User Intent to Assist Goal-oriented Social Search. In *Proceedings of the ACM Workshop on Search in Social Media*, California, USA: ACM Press pp. 35—42.

Strohmaier, M., Prettenhofer, P., and Lux, M. (2008). Different Degrees of Explicitness in Intentional Artifacts - Studying User Goals in a Large Search Query Log. In *Proceedings of the CSKGOI'08 Workshop on Commonsense Knowledge and Goal Oriented Interfaces*, Spain.

Strohmeier and Kröll (2012). Acquiring knowledge about human goals from Search Query Logs. Information Processing & Management 48(1), Elsevier, pp. 63-82.

Sun, X., Sharples, S. & Makri, S. (2011). A User-Centred Mobile Diary Study Approach to Understanding Serendipity in Information Research. *Information Research*, 16(3), paper 492. Available at: http://InformationR.net/ir/16-3/paper492.html

8. Appendix I: Prototype Goal Taxonomy

1: Goals	
1.1: Generic Goals	
	1.1.1: NOTE STH
	1.1.2: TRY/ATTEMPT TO DO STH
	1.1.3: INTEND TO DO STH
	1.1.4: WANT TO DO STH
	1.1.5: ENCOURAGE STH
	1.1.6: BE QUALIFIED IN STH
	1.1.7: OBTAIN STH
	1.1.8: LOOK AT STH
	1.1.9: ATTEND STH
	1.1.10: PURCHASE STH
	1.1.11: ENGAGE IN STH
	1.1.12: LISTEN TO STH
	1.1.13: MENTION STH
	1.1.14: ASK SOMEONE ABOUT STH

	1.1.15: EXPERIENCE STH
	1.1.16: FINISH STH
	1.1.17: DESIRE STH
	1.1.18: NOTICE STH
	1.1.19: UNDERSTAND STH
	1.1.20: VISIT PLACE
	1.1.21: DECIDE STH
	1.1.22: IN NEED OF STH
	1.1.23: WORK ON STH
	1.1.24: COMMENCE STH
	1.1.25: FOCUS ON STH
	1.1.26: HELP PERSON
	1.1.27: GO TO PLACE
1.2: Domain Specific Goals	
1.2.1: Research Domain	
1.2.1.1: Analyse/Synthesise	
	1.2.1.1.1: CONNECT ENTITIES
	1.2.1.1.2: CREATE STH
	1.2.1.1.3: INVESTIGATE STH
	1.2.1.1.4: MAP STH
	1.2.1.1.5: MODEL STH
	1.2.1.1.6: TEST STH
	1.2.1.1.7: COMPARE STH WITH STH
	1.2.1.1.8: DEFINE STH
1.2.1.2: Communicate / collaborate	
	1.2.1.2.1: RECOMMEND STH
	1.2.1.2.2: MEET PEOPLE
	1.2.1.2.3: TALK TO PERSON (TalkTo)
	1.2.1.2.4: DISSEMINATE STH
	1.2.1.2.5: TELL SOMEONE ABOUT STH
	1.2.1.2.6: CONTACT ENTITY
	1.2.1.2.7: SHARE STH
	1.2.1.2.8: SUGGEST STH
	1.2.1.2.9: COLLABORATE WITH ENTITY
	1.2.1.2.10: DISCUSS STH
1.2.1.3: Develop/devise	
	1.2.1.3.1: SOLVE STH
	1.2.1.3.2: DEVELOP STH
	1.2.1.3.3: BUILD STH
	1.2.1.3.4: MAKE STH
1.2.1.4: Find/search	
	1.2.1.4.1: FILTER STH
	1.2.1.4.2: FIND STH
	1.2.1.4.3: BROWSE STH
	1.2.1.4.4: LOOK UP STH
1.2.1.5: Implementing	
	1.2.1.5.1: USE STH
	1.2.1.5.2: EXTEND STH
	1.2.1.5.3: APPLY STH
1.2.1.6: Producing outcomes	
	1.2.1.6.1: WRITE STH
	1.2.1.6.2: DESIGN STH
	1.2.1.6.3: ORGANIZE STH
	1.2.1.6.4: FACILITATE STH
1.2.1.7: Understand	
	1.2.1.7.1: CONSIDER STH
	1.2.1.7.2: READ STH
	1.2.1.7.3: STUDY STH
	1.2.1.7.4: LEARN STH
2: Goal Indicators	
	2.1: PLAN TO DO STH
	2.2: TRY/ATTEMPT TO DO STH
	2.3: INTEND TO DO STH
	2.4: WANT TO DO STH

Building a Baseline Supervised Relation Extraction System Using Freely-Available Resources

Stefan Daniel Dumitrescu
Research Institute for Artificial Intelligence, Romanian Academy
Bucharest, Romania
E-mail: sdumitrescu@racai.ro

Abstract

The article presents an easy-to-follow guide to building a supervised Relation Extraction system using free resources. The reader can see how to build the system in a step-by-step fashion, what tools, methods and data sources are needed and how they can be processed and then used, as well as see the practical results of such a system. Also, we explore the surface of performance evaluation giving the reader some basic measures and definitions, like: binary classifiers, cross-validation, feature space with different features extracted from annotated sentences, impact of features in different classifiers, confusion matrices and feature evaluation methods.

Keywords: supervised, relation extraction, flat feature space, SVM classifier, baseline system, how-to guide

1. Introduction

Almost all papers that propose new Relation Extraction (RE) systems have to compare with the results obtained using a standard yet state-of-the-art baseline system. However, few of those papers actually describe in sufficient detail how to build such a baseline system. The aim of the article is not to present a high performance Relation Extractor, but to show the relative novice in this field how to build such a classic baseline system: where to extract data from, how to clean and process the data, how to build classifiers trained on the extracted data and how to evaluate their performance using a free but powerful data mining tool. The basic concepts common in the RE field (and not only) are also introduced.

We make the following design decisions / constraints:

- the system is designed to correctly classify relations and not to perform relation detection (meaning we attempt to classify a relation when given an example that is known to hold an actual relation);
- the system will handle a limited number of pre-defined relations (proof-of-concept);
- we use an independent data source;
- we use supervised classifiers – we train one Support Vector Machine (Cortes & Vapnik, 1995) classifier per relation;
- we use a specific set of features extracted from the data set;
- we restrict the task of RE only to binary relations between named entities; also, we consider the task of NER - Named Entity Recognition (Grishman & Sundheim, 1996) as solved (100% accurate).

Section 2 describes related work focusing on feature-based classifiers. In Section 3 we describe in detail the individual building steps, while in section 4 we evaluate the system's results and comment upon its characteristics. We draw a few conclusions in the final section 5 of the paper.

2. Related Work

At its core, Relation Extraction is basically a classification problem: given a pair (or tuples of several entities) we need to detect whether there is a relation between the entities and what that relation might be. Currently, supervised approaches have the best performance in relation classification. From the point of view of the classification methods, we can separate the feature-based vs. tree and graph kernel methods. The majority of classifiers are feature-based; however some of them (like the SVM or the Perceptron) can be extended to work with kernel functions.

Feature-based classifiers use labeled examples for training; each of these examples is represented as an n-dimensional array where each dimension is a feature. A feature can be boolean (true/false), numeric (any real number), nominal (for example given a number of examples, every distinct value for a nominal feature can be seen as a different class) or even directly strings. While feature-based classification is relatively intuitive, the problem with it is that having a high dimensionality space (having a large number of features) leads to computational issues. Kernel methods are designed to solve this problem by bypassing the need for explicit representation of feature vectors. At their core, kernel methods are similarity functions that, given two objects (examples or instances), will output a similarity score (Cristianini & Shawe-Taylor, 2000). The most important property of a kernel function is that the product or sum of kernels is a kernel itself.

Currently, the best results in relation classification are obtained using supervised classifiers with custom tree kernels (Zhou, Zhang, Ji, & Zhu, 2007). However, every such system is compared with a baseline, typically under the form of feature-based classifiers. We further focus on this latter type of classifiers.

The performance of feature-based classifiers can be improved by mainly two methods: better features and better classification algorithms. While improving the classification algorithms is a rather difficult mathematical challenge, with current methods such as SVMs, Voted Perceptron (Freund & Schapire, 1999), different variants of hierarchical classification, etc., showing good performance, the search for better features is an open field. Among the first important feature-based systems is the proposal of Zhao & Grishman (2005) where they have used tokenization (using sequences of n-grams), parse and dependency trees, and also a combination of these features. Systems that followed attempted to integrate increasingly more features. In the same year another

system proposed by Bunescu & Mooney (2005) introduced the shortest path in the dependency tree as a required information that asserted that two entities were in a relation. The developed kernel thus incorporated words and word class features of the path components.
Newer systems expanded the feature space even more. Several new feature types were used, including part of speech tags, entity subtype, entity class, entity role, semantic sentence representation and also using the WordNet synonym sets. The system implemented by Wang, Li, Bontcheva, Cunningham, & Wang (2006) is a good example of knowledge engineering applied to feature-based classifiers, having extracted almost 100 different features.
The word based features include the entities themselves, bigrams before and after the entities, the heads if available, etc. part of speech (POS) tags are used similarly to word features (the main argument is that words are too different / too sparse compared to their associated POS tags). Entity features include their type and subtype (based on the ACE 2004 [1] classification). Among the sentence-related features we can name the number of words separating two entities, the number of other entities between the two entities, etc. Several combinations of entity features and sentence features have been created to obtain features that better discriminate examples in cases of sparse sentence/entity features. Syntactic and dependency features were also used, such as chunks obtained by parsers (noun phrases - NP, verb phrases - VP, etc.), whether the two entity mentions are included in the same NP/VP, the type and voice information of the VP, combination of the head words and their dependent words for the two entity involved, the combination of the dependency relation type and the dependent word of the heads of the two entities, the path of dependency relationship labels connecting the heads of the two entities. Last but not least, WordNet (Miller, Beckwith, Fellbaum, Gross, & Miller, 1990) features have been used: use the ID of the first synset (a synset is a list of synonyms having a certain meaning) for the entities themselves, the words surrounding them and linking them. To avoid performing Word Sense Disambiguation the most frequent synset - which is by design the first - is always used.
Zhou, Zhang, Ji & Zhu (2007) proposed a system that combined feature-based and tree kernel-based methods in a way that they complemented each other. Also their system is amongst the first to show that an individual tree kernel can achieve better performance than the state-of-the-art linear kernel.
In this paper we aim at using a subset of the possible feature-space described so far, explain how to obtain these features and how to integrate them into a working classification system. For the evaluation of the performance we will use the freely-available WEKA (Hall, et al., 2009) data mining solution, with SVM with the standard polynomial kernel as our classification algorithm choice. While the inner working of the SVM and more specifically the polynomial kernel is out of the scope of this paper, a short working summary is that the SVM, given a number of n-dimensional points (the point is an analogue for an example consisting of n attributes / features and its class) attempts to find the largest margin that separates the positive and negative examples on each of the n dimensions thus constructing a separation hyper-plane. New examples are then mapped into the same n dimensional space and are predicted to belong to the category which represents the side of the plane that they fall into. There are a number of good tutorials available online[2].

[1] http://ace2004.isr.ist.utl.pt/

3. Building the System

Any supervised RE system has two major components: 1. data acquisition and 2. using this data to build the classifier. We discuss the data acquisition phase (data extraction, cleanup and annotation – sections 3.1 - 3.3) and then the classifier training (section 3.4).

3.1. Data Extraction

In any machine-learning algorithm, the amount and quality of the data provided makes the biggest difference in classification performance.
Because we do not intend to build a specialized system (such as those participating in ACE[3] type competitions), but a general baseline that can be later customized, we choose freely-available sources of information to construct our data source. As such, for the pre-defined relations and associated seed-pairs we use the YAGO ontology (Suchanek, Kasneci, & Weikum, 2007) and for the actual sentences that comprise the training data we use the Web. YAGO, standing for Yet Another Great Ontology is an automatically constructed, high accuracy (95%+) ontology based on Wikipedia and WordNet. Currently at its second version, the core package contains 2.6 million entities and about 33 million facts[4].
The data extraction process is the following:
Step 1. We identify a number of relations that we want to extract. In this paper we investigate 5 relations: `bornIn`, `diedIn` and `isLeaderOf` (relations between a person and a location), `locatedIn` and `hasCapital` (relations between a location and another location). We use these specific relations as we attempt to compare the results with relations that have the same domain and value range).
Step 2. For each relation we create a list of entity pairs (E_1, E_2) also named "seed pair". The entities are known to stand in the chosen relation (E_1 `relation` E_2). For example, for relation `bornIn`, a pair of entities is (Einstein, Ulm), as we know that Einstein was born in the town of Ulm, Germany. This list is created interrogating YAGO about the entities it knows about that stand in a particular relation. YAGO can be used in many formats, from plain text to xml files to SQL databases. We choose to use the SQL format as it offers the best performance short of loading the entire ontology into system memory. For example, to obtain the first 1000 entities that stand in the `bornIn` relation, we write the following SQL query: "`SELECT arg1, arg2 FROM facts WHERE relation = 'bornIn' LIMIT 1000`". We will obtain a table with two columns, on each row having a person (arg1) that was

[2] http://www.svms.org/tutorials/
[3] Automatic Content Extraction Conference/Competition : http://projects.ldc.upenn.edu/ace/
[4] YAGO download and details at: http://www.mpi-inf.mpg.de/yago-naga/yago/downloads.html

born in a location (arg2).
Step 3. Having obtained the seed pairs, we now have to obtain the actual sentences that contain these pairs. The intuition is that searching the web for sentences that contain the entities in each seed pair we will obtain a sufficient number of examples, both positive and negative.
As such, we use the Bing[5] search engine to obtain the initial list of sentences. We chose Bing because it allows searching for the two entities within a definable maximum window[6]. To speed up the development of the web parser, we use the Bing API. This is an interface that given an input search query, it returns an xml document containing the search results.
For example, a valid query for seed pair (Kidangoor, Kerala) that stand in the `locatedIn` relation would be:

```
http://api.bing.net/xml.aspx?AppId=8236FA820E20
281959CB9CEFE09&Version=2.2&Market=en-US&
Query=\"Kidangoor\"+near%3A7+\"Kerala\"&Sources
=web&Web.Count=3
```

This places a query to page api.bing.net/xml.aspx, with the following parameters: AppId (a static id obtained from Bing that is needed to access the API), Version (the version of the XML service interrogated, here *2.2*), Market (we wish only for English results so we need to specify the market as being *en-US*), the Query containing a seed pair – our two entities Kidangoor and Kerala (that stand in the `locatedIn` relation, as the town of Kidangoor is located in the state of Kerala, India) with the Bing keyword *near:7* between them, forcing results that contain the two entities in a window of maximum 7 words, the Sources parameter (stating that we request *web* results – can be images/news/video etc), ending with the Web.Count parameter (specifying how many results we want for our query; here we request the top *3* results). The chosen values (window of 7 words and top 3 results only) were heuristically chosen as we obtained good result diversity with them. As a response to our query, we are presented with an xml page that contains WebResult elements:

```
<web:WebResult>
  <web:Title> College of Engineering Kidangoor …
</web:Title>
  <web:Description> College of Engineering
Kidangoor is located in Kottayam District of Kerala.
It was set up in the year
2000-2001. …</web:Description>
  <web:Url>
http://www.highereducationinindia.com/institute
s/college-of-engineering-kidangoor-21.php
</web:Url>
  <web:DisplayUrl>
www.highereducationinindia.com/
...of-engineering-kidangoor-21.php
</web:DisplayUrl>
<web:DateTime>2012-01-29T12:27:14Z</web:DateTim
e>
</web:WebResult>
```

It can be seen that there are several interesting elements. We could follow the link in the Url element and parse the web page until we find the two entities in the same sentence. However, if we focus our attention on the Description element we can see that it contains the descriptive text snippet below the result link. This snippet is guaranteed to contain the entities in the seed pair. Considering that it is much faster to directly load this snippet text than it is to parse the web page it came from, we choose to use it directly – store it as a representative sentence containing the seed pair entities in the current relation.
For example, the process of extracting the initial data set for relation `bornIn` is: 1. Select `bornIn` as the current relation; 2. query a data source (here, YAGO) of entity pairs that stand in that particular relation and obtain a list of n pairs; 3. for every entity pair ask a search engine (here, Bing) for the first k top results (here, k=3) using the predefined search query, and save each result obtained in a list with the generic format (E_1, E_2, result_snippet). It must be noted that the obtained list contains both positive and negative examples of the relation – the search engine simply returns sentences in which the seed entities are close to each other – this does not mean that the words linking the entities are guaranteed to form a positive example. So, with this single pass we extract both positive and negative examples for each of the targeted relations (at this point we do not know if a result snippet is positive or negative – it will be manually decided in step 3.4).

3.2. Data Cleaning

Because we have used a 'noisy' data source – the Web – presented through the interface of a search-engine, a cleaning step is required to identify and remove bad candidate sentences. Bad candidate removal does not mean removing negative examples of the relation but removing examples that cannot have features correctly extracted from, as further detailed.
The initial cleaning starts with a sentence detector applied on every text snippet extracted. As the text snippets are automatically generated by the search engine to highlight the query terms, there is no guarantee that both terms will appear in one sentence. Sometimes the terms appear in different sentences in the snippet, sometimes the sentences do not have a beginning or end (signalled by the automatically appended/prepended "…" punctuation) or sometimes the three-dot sign is found right between the terms. To detect such cases we used the sentence detector provided by OpenNLP[7] tools package. The detector is based on a trained Maximum-Entropy model that is able to detect sentence boundaries with high accuracy. If the two terms are found in a single sentence contained in the text snippet, only that sentence is further kept. All other sentences and snippets that fail this check are discarded. Also, at this point, to enforce non-duplicate data, a hash-table of valid sentences is kept.
After all sentences have passed through the sentence detector, the last part of the cleaning process involves detecting similar sentences. This check is needed because the Web contains vast amount of not only duplicate but also similar data. Using the JaroWinkler (Winkler, 1990) string similarity algorithm (algorithm available in free

[5] Bing Search Engine: http://www.bing.com
[6] http://msdn.microsoft.com/en-us/library/dd251056.aspx
[7] http://incubator.apache.org/opennlp/index.html

string similarity Java libraries[8]) we compute the similarity score between each sentence and the last *n* sentences (we chose *n* = 9). Having heuristically determined a similarity threshold value of 0.8, every sentence that scores higher is dropped, meaning that we already have a very similar sentence in the data set. We cannot compute the similarity between a sentence and all of the other sentences because that would be too time-consuming (quadratic vs. linear complexity), and most often similar sentences are found in the last 3 results processed (as for each seed pair we extract the top 3 results, giving a high probability of duplicate/similar data). Using a simple queue of the last checked sentences we obtain linear performance for this cleaning step.

We observed that the cleaning step, depending on the relation being cleaned, drops between 30%-60% of extracted sentences. However, given that the ontology can provide literally thousands of seed pairs and for practically all of them we can obtain several web search results, the amount of sentences lost due to cleaning is not an important factor.

3.3. Data Annotation

As specified in the Introduction and Related Works sections, we choose to build a supervised baseline system implementing flat features for a SVM classifier. To extract the features we process each sentence using hand-crafted rules as well as using Stanford's CoreNLP (Klein & Manning, 2003) tool to obtain syntactic and dependency trees. Syntactic trees look similar to dependency trees (dependency grammar is equivalent to constituency grammar if there is one restriction of the constituency grammar – that in each phrase a word is set to be its head (Gaifman, 1965)) and in some cases the NLP field treats both tree types the same (Covington, 2000). A dependency tree makes explicit relationships between words in terms of heads and dependents (see figure 1) while a syntactic tree makes explicit syntactic constituents visible in a sentence (see figure 2).

Let's take the following sentence as an example: "Muppet creator *Jim Henson* was born in the city of *Greenville*.", with Jim Henson as E_1 and Greenville as E_2.

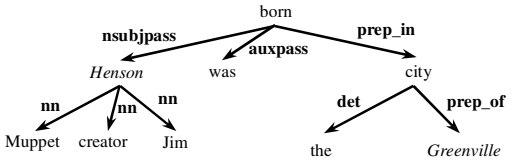


Figure 1. Collapsed dependency tree example

For example, the *nsubjpass* [9] relation means that there is a passive subject – predicate relation between governor word *born* and dependent word *Henson*.

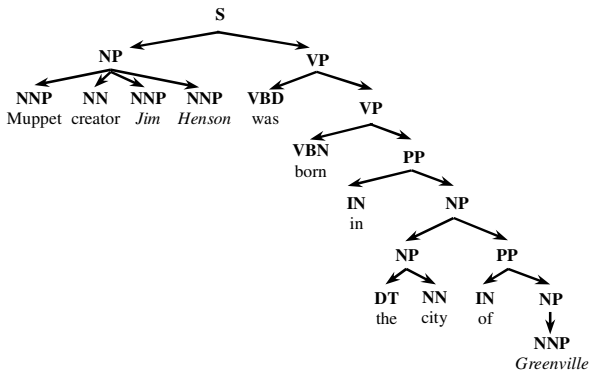


Figure 2. Syntactic tree example

The syntactic tree shows how the words are grouped into bigger and bigger noun phrases (NP), verb phrases (VP), and others types of phrases (PP, etc.), up to the sentence root (S).

From all possible features we can extract, we choose a limited subset that includes features that come from the sentence itself, from the part of speech tags and also from the syntactic and dependency trees. We present the extracted features:

F0 - Entity Order. We identify if the first entity in the seed pair appeared before the second. We mark the feature with a Boolean true/false value:

F1 – String between the entities. We extract the substring between the two entities, not including them.

F2 – Word Count between Entities. We count the number of words that separate the entities.

F3 – POS Tag String. As we have already identified the POS tags of every word, we append together the tags of the substring between the entities.

F4 – Dependency Relation Path. The Stanford Parser offers the dependency tree as a directed graph. We identify the shortest path between the entities and we 'collapse' the path as a string.

F5 – Verb. We check to see if there is a verb linking the two entities. We initially search for a verb between the two entities. If we do not find a verb there, we search for the verb in the dependency tree, as it may reside before the first entity or after the second one. We use *null* to mark a verb-less relation.

F6 – Syntactic Path between E_1 and the Verb. This time using the syntactic tree we identify the path between the first entity and the verb. We mark the change of height in tree with ">" as up and "<" as down. If F5 is *null*, then F6 will also be *null*.

F7 – Syntactic Path between the Verb and E_2. Similar to F6, only this time we search for the path starting from the Verb to E_2.

Table 1 lists the features extracted for the previous example sentence "Muppet creator *Jim Henson* was born in the city of *Greenville*.". For every relation we thus extract the sentences' features and, for ease of further usage, we store them in both a serialized format (using Java serialization) and in the *.arff* files format (WEKA CSV-like (Comma Separated Values) text format).

[8] http://sourceforge.net/projects/simmetrics/files/simmetrics _jar/

[9] Dependency relations are explained at : http://nlp.stanford.edu/software/stanford-dependencies.shtml

Feature	Value
F0	true
F1	"was born in the city of"
F2	6
F3	"VBD VBN IN DT NN IN"
F4	"nsubjpass prep_in prep_of"
F5	"born"
F6	"NNP > NP > S < VP < VP < VBN"
F7	"VBN > VP < PP < NP < PP < NP < NNP"

Table 1. Feature example

3.4. Building the classifier

Because we aim to build a binary classifier for every relation, we need a way to quickly annotate a sub-set of the extracted sentences. We created a simple text-based Java program that, for every relation, reads the first *n* sentences and sequentially displays them along with their features on-screen. The annotator is required to press a button ('y' or 'n' key) to specify if that sentence is a positive or negative instance of the respective relation. We know that, for each relation, the entities obtained from YAGO actually stood in that relation. We have to determine only if the extracted sentences that contain these entities represent positive or negative instances. For example, for entities *Einstein* and *Ulm*, sentence "*Einstein* was born in *Ulm*" is a positive instance while "*Einstein* went to primary school in *Ulm*." is a negative instance.
This allows the user to process a sentence every 2-3 seconds. We have processed 700 sentences in this manner for each relation (thus taking about 20-30 minutes per relation). This manual method of annotation is not scalable to large corpora as it requires time to annotate large numbers of instances as well as having to consider the inter-annotator-agreement (the case where, when using multiple annotators, they do not agree with each other). However, for building a baseline system with good performance, a limited number of annotated instances is acceptable.
The Java program writes the sentences directly in .arff format using the WEKA API. It should be noted that each relation is kept separate. We do not use positive instances of one relation as negative instances of another. The extracted and cleaned data for each relation contains both negative and positive instances for that relation only.
Table 2 presents a few examples of positive (+) and negative (-) instances of the `bornIn` relation. The entities are italicized and also marked with E_1 and E_2 to highlight their order in the sentence. The first and last (sixth) example show the difficulty of identifying relations. For people, whenever a name is followed by an opening parenthesis, we expect some biographical information. In the first (positive) example Ray Mercer is followed by Jacksonville, which we assume is his birth place (even though we have no other context besides the sentence itself), while in the last (negative) example simply putting a comma and then a location (London) next to a person's name does not necessarily mean that the person was born there, it could simply show his location at a particular date, like the ending of a news report which specifies the name of the reporter and place of the cast. In this types of scenarios, the classifier has to rely only on two different features: *F1*="(" & *F3*="-LRB-" for the positive example (LRB is the encoding for left regular brace) vs. *F1*="," & *F3*="COMMA" for the negative example, as all other features are identical.

+/-	Sentence
+	200 POUNDS: Gold -- *Ray Mercer*[E1](*Jacksonville*[E2], Fla.).
+	*Emile Berliner*[E1] was born in *Hanover*[E2], Germany in 1951.
+	A *Vancouver*[E2] native *Teryl Rothery*[E1] always knew she wanted to be an entertainer.
-	Famous persons from *Watervliet*[E2] include *Joe Alaskey*[E1].
-	Born *James Larkin*[E1] Jones, the son of a *Liverpool*[E2] docker, he worked in the docks himself for some years.
-	*Tony Newton*[E1], *London*[E2], United Kingdom.

Table 2. Positive and negative examples for the `bornIn` relation

At this point, we are ready to train the first binary classifier. Using either the WEKA API or WEKA software directly, we load the arff file containing the annotated sentences for a relation. As the features (with the exception of *F0* which is Boolean and *F2* which is numeric) are strings, we convert them to nominal classes to be able to use the Polynomial Kernel of the SVM classifier. We could use the string kernel directly, but that would lead to rather poor performance (on average, under 60% classification accuracy). As such, we build for every relation a SVM model.
Table 3 presents the number of positive/negative examples and the cross-validation accuracy per relation. The table provides an interesting insight on the diversity of sentence instances and the difference in difficulty to obtain, for example, an equal number of positive examples for each relation. While the search query stays the same (E_1 and E_2 near to each other), there are many more positive examples of the `bornIn` relation (48%) than `diedIn` (only 20.1%) in the same number of extracted sentences; the difference is even more accentuated for the `locatedIn` (75.8%) vs. `hasCapital` (12.4%) relations, even though `hasCapital` is basically a subset of `locatedIn` (any relation that states that a city is a capital of a country also means that that city is located within the respective country).

Relation	Positive Examples	Negative Examples	10 fold cross-val.
`bornIn`	336 (48%)	364 (52%)	85.5%
`diedIn`	141 (20.1%)	559 (79.9%)	92.1%
`isLeaderOf`	399 (57%)	301 (43%)	79.4%
`locatedIn`	531 (75.8%)	169 (24.2%)	85.5%
`hasCapital`	87 (12.4%)	613 (87.6%)	92.7%

Table 3. Relation class distribution and cross-validation classifier accuracy on each 700 sentence hand-annotated set

The cross validation figure shows how well the classifier was trained on a particular data set (this feature is available directly in WEKA). The cross-validation means that the classifier is trained on a fraction of the available instances and then tested on the remaining fraction. 10-fold means that we 'cut' the data-set in 10 'folds' or

fractions. We train on the first 9 folds (90% of the data) and test on the remaining fold (10%). Also, to ensure that the results are not biased by choosing a single fold, this process is automatically repeated 10 times, each time keeping a different fold of the data for testing.

At this point we have constructed the SVM binary models for each of our targeted relations. The trained classifiers can be exported as a binary file and loaded using the WEKA Java API directly in an application. So, given a sentence with identified entities and annotated in the same manner (with the same features), we can recognize relations using our trained classifiers.

4. Evaluation of the System

For the evaluation of the system, we must first specify that we measure only precision, meaning whether a given annotated sentence is found to be a positive or a negative instance of a certain relation. We do not measure recall (recall means correctly identifying that there is a relation between two entities). Because in this article we focus on building the relation classification component, we assume that the Named Entity Recognition (NER) module used to identify entities in a sentence is 100% percent accurate and that there is always a relation between the given entities, whether a positive or a negative relation. The choice of ignoring recall allows us to simplify the design of the system and also to evade the NER's inherent errors. Using directly the entities provided by the ontology we can ensure that they are identified with 100% accuracy. For this reason we did not include in our feature space the type of the named entities. For example, for the `bornIn` relation we could have included a new *F8* feature as equal to "person" and *F9* as "location", as `bornIn` is a relation defined over the *person* domain with values in the *location* domain. However, as we test either individual relation classifiers (binary or yes/no classifiers) or multi-class SVM classifiers trained on relations defined on the same domain/range (which, at their core, are binary classifiers arranged in different configurations, e.g. one-vs-all, one-vs-one, etc), the use of NER-related features like *F8*/*F9* would have been redundant.

Having our 5 relations, we can create 2 multi-class classifiers: one defined over person →location including three relations: `bornIn`, `diedIn` and `isLeaderOf`, and one defined over location→location including the remaining two relations: `locatedIn` and `hasCapital`. The first multi-class classifier is built to see how relations that are similar in nature (`bornIn` vs `diedIn`) are discriminated and correctly identified, and also to see how the introduction of a relatively distinct relation (`isLeaderOf`) impacts overall classification accuracy. The second multi-class classifier is built to evaluate how two similarly defined relations are identified, where one relation (`hasCapital`) is actually a logical subset of the other (`locatedIn`).

Using only the positive instance of each relation, we construct the two multi-class classifiers. Example: we take the positive examples of `bornIn` that will represent the `bornIn` class; similarly for `diedIn` and `isLeaderOf`. The classifier will now have to choose between one of the three possible classes for an unknown sentence.

The results show very good classification accuracy in our restricted training/test set. Using the same 10-fold cross-validation method, we obtain a 91.5% classification precision for the three class classifier and 94% for the two class classifier. This accuracy is obtained in spite of the class imbalance (class skew): for the first three-class classifier we have 399 instances for `isLeaderOf`, 336 for `bornIn` and only 141 for `diedIn`. The even more accentuated class skew for the second two-class classifier doesn't seem to affect performance very much.

We present the confusion matrix for both classifiers. Such a matrix shows how many instances have been correctly classified for each class, and if incorrectly classified, in which class were they classified into.

```
=== Confusion Matrix ===
   a   b   c   <-- classified as
 381  16   2 |   a = isLeaderOf
  49 284   3 |   b = bornIn
   6   3 132 |   c = diedIn

=== Confusion Matrix ===
   a   b   <-- classified as
 517  14 |   a = locatedIn
  17  70 |   b = hasCapital
```

Figure 3. Confusion matrices for both multi-class classifiers

Looking at the confusion matrix for the first classifier we can draw an interesting conclusion: even if the `bornIn` and `diedIn` relations appear similar, the classifier was able to correctly classify almost all instances of the relations. It actually misclassified many more instances of `bornIn` as `isLeaderOf` (49) than `diedIn` (only 3). Almost all instances of `diedIn` were correctly classified, only 3 as `bornIn` and 6 as `isLeaderOf`.

For the confusion matrix of the second classifier we can see that the error rate is also very small, most instances being correctly classified. However, even from the simple example of two-class vs. three-class classifier we can see that an *n*-class classifier will perform increasingly worse as *n* gets larger.

Another interesting aspect to analyze is the features themselves: what are good/bad features, how a feature influences the classifier, and so on. WEKA offers a number of different methods to analyze the feature space.

Value	Feature
0.5108	*F5 – Verb*
0.4486	*F3 – POS Tag String*
0.4424	*F4 – Dependency Relation Path*
0.4022	*F1 – String between the entities*
0.3532	*F7 – Syntactic Path between the Verb and* E_2
0.2421	*F6 – Syntactic Path between* E_1 *and the Verb*
0.1376	*F0 - Entity Order*
0.0507	*F2 – Word Count between Entities*

Table 4. Feature ranking using the Relief Evaluator for the three-class classifier

To perform our analysis we choose the Relief Attribute Evaluator (Robnik-Sikonja & Kononenko, 1997) (WEKA calls features attributes). This measure evaluates the worth of an attribute by repeatedly sampling an instance and considering the value of the given attribute for the nearest instance of the same and of different classes.

Using a ranking algorithm (directly implemented in WEKA) in conjunction with the Relief Attribute Evaluator method configured to sample all instances and with a 10 nearest-neighbor maximum limit (instances have been already randomized to minimize bias) we obtain the ranking presented in table 4.

We can see that for the three-class classifier the most important feature was the verb, followed by the POS tag string and the dependency path. Entity order was not really important and almost non important was the word count between entities.

Value	Feature
0.6875	*F3 – POS Tag String*
0.6778	*F1 – String between the entities*
0.6578	*F4 – Dependency Relation Path*
0.0448	*F2 – Word Count between Entities*
0.033	*F6 – Syntactic Path between E_1 and the Verb*
0.0314	*F0 - Entity Order*
0.023	*F7 – Syntactic Path between the Verb and E_2*
0.0206	*F5 – Verb*

Table 5. Feature ranking using the Relief Evaluator for the two-class classifier

In table 5 we see that the feature importance has changed. Here, the verb has fallen directly to the last position, as the most uninformative feature. This is actually to be expected, as most `locatedIn` relations are in the form of "*Paris*, *France*" while `hasCapital` relations are in the form of "*Jijiga*, the capital of *Somali Region* ...", both having no verbs linking them.

5. Conclusions

This article is meant to be taken as a practical introduction to relation extraction where the reader can see how to build a standard supervised RE system in a step-by-step fashion, what tools, methods and data sources are needed and how they can be processed and then used, as well as see the practical results of such a system. Also, we explore the surface of performance evaluation giving the reader some basic measures and definitions, like: binary classifiers, cross-validation, feature space with different features extracted from annotated sentences, impact of features in different classifiers, confusion matrices and feature evaluation methods.

Because we have designed the system as modular - meaning that we targeted only relation classification and not relation detection, and have kept the system data as serialized objects (annotated sentences as binary objects and WEKA arff data files, binary classifier models, etc.), we can quickly create custom models, multi-class SVMs or other classifiers in order to extend and customize the system to conform to any baseline requirements.

6. Acknowledgements

The work reported here was funded by the project METANET4U by the European Comission under the Grant Agreement No 270893.

7. References

Bunescu, R., & Mooney, R. (2005). A shortest path dependency kernel for relation extraction. *Proceedings of the Joint Conference on Human Language Technology / Empirical Methods in Natural Language Processing (HLT/EMNLP)* (p. 724{731). Vancouver: Association for Computational Linguistics.

Cortes, C., & Vapnik, V. N. (1995). Support-Vector Networks. *Machine Learning, 20*(3), 273-297.

Covington, M. A. (2000). A Fundamental Algorithm for Dependency Parsing. *39th Annual ACM Southeast Conference*, (pp. 95-102).

Cristianini, N., & Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines.* Cambridge: Cambridge University Press.

Freund, Y., & Schapire, R. E. (1999). Large margin classification using the preceptron algorithm. *Machine Learning*, 277-296.

Gaifman, H. (1965). Gaifman, Haim. In *Information and Control* (pp. 304-307).

Grishman, R., & Sundheim, B. (1996). Message Understanding Conference - 6: A Brief History. *International Conference on Computational Linguistics.*

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations, 11*(1).

Klein, D., & Manning, C. D. (2003). Accurate unlexicalized parsing. *ACL.*

Miller, G. A., Beckwith, R., Fellbaum, C. D., Gross, D., & Miller, K. (1990). WordNet: An online lexical database. 235-244.

Robnik-Sikonja, M., & Kononenko, I. (1997). An adaptation of Relief for attribute estimation in regression. *Fourteenth International Conference on Machine Learning*, (pp. 296-304).

Suchanek, F. M., Kasneci, G., & Weikum, G. (2007). Yago - A Core of Semantic Knowledge. 16th international World Wide Web conference (WWW 2007).

Wang, T., Li, Y., Bontcheva, K., Cunningham, H., & Wang, J. (2006). Automatic Extraction of Hierarchical Relations from Text. *Proceedings of the Third European Semantic Web Conference (ESWC 2006).* Budva.

Winkler, W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. *Proceedings of the Section on Survey Research*, (pp. 354--359). Wachington, DC.

Zhao, S., & Grishman, R. (2005). Extracting relations with integrated information using kernel. *ACL 2005 Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, (pp. 419-426). Ann Arbor, Michigan.

Zhou, G., Zhang, M., Ji, D., & Zhu, Q. (2007). Tree Kernel-Based Relation Extraction with Context-Sensitive Structured Parse Tree Information. *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, (pp. 728-736).

Arguments for Phrasal Verbs in Croatian and Their Influence on Semantic Relations in Croatian WordNet

Daniela Katunar, Matea Srebačić, Ida Raffaelli, Krešimir Šojat
University of Zagreb, Faculty of Humanities and Social Sciences
Ivana Lučića 3, Zagreb, Croatia
E-mail: dkatunar@ffzg.hr, msrebaci@unizg.hr, ida.raffaelli@ffzg.hr, ksojat@ffzg.hr

Abstract

In this paper we introduce the category of phrasal verbs in Croatian lexicon and grammar description in order to show their influence on semantic relations, namely synonymy and polysemy in Croatian WordNet (henceforth CroWN). We discuss the practical and theoretical implications that arise from the introduction of the category of phrasal verbs in the description of the Croatian lexicon. We also address the interaction of synonymy and polysemy as manifested in the semantic relations of phrasal verbs to their monolexemic counterparts and facilitated by the structure of CroWN. The lemmatization of phrasal verbs in Croatian dictionaries and its modification for purposes of improving semantical relations in CroWN is also discussed. We also propose building of the Croatian phrasal verbs database, describe its structure and its further expanison which would facilitate extraction and incorporation of phrasal verbs into CroWN, and thus improve MT systems and information extraction via this computational lexical resource.

Keywords: phrasal verbs, semantic relations, synonymy, polysemy, Croatian WordNet

1. Introduction

Synonymy and polysemy are ubiquitous lexical semantic relations that continously structure the lexicon of a language. However, when it comes to their enumeration and notation within lexical resources, one is often faced with many caveats as to their valid representation. Particularly with regards to polysemy, the main problem seems to be a precise enumeration of various senses of a polysemous lexical unit, as well as their disambiguation from the various contexts they appear in (see Fellbaum, 2000, Fillmore & Atkins, 2000). On the other hand, though synonymy has been well described via thesauri as a very salient lexical relation, there is rarely an opportunity to study and represent the interaction of synonymy and polysemy within the format of tradicional dictionaries (see also Fellbaum, 1998). A fertile testing ground for such studies seems to be within the format of conceptual lexica such as WordNet. Since WordNet is conceived and built as complex network of lexical-semantic relations, it has a structure that necessitates the incorporation of various lexical-semantic relations, such as synonymy, antonymy, polysemy and hyperonymy/hyponymy in unison, i.e. it makes explicit their connections cross-cutting the structure of the lexicon of a language.

For instance, a polysemous unit in the Croatian WordNet (henceforth CroWN) *masa* 'mass' has seven distinct senses, three of which are *masa:1* 'a physical unit of weight', *masa:2, svjetina, puk, gomila* 'a crowd of people' and *vodena masa:3, vodena površina* 'lit. water mass, a body of water'. As the examples show, there is a three-way distiction between the senses in the way they interact with their surrounding lexical units. *Masa:1* 'a physical unit of weight' is a standalone lexical unit having its own synset which denotes the source (or basic) meaning of 'mass' in general, that of weight. Conversely, *masa:2* is related to other lexical units in the same synset *svjetina, puk, gomila* 'a crowd', which clearly indicate the metaphorical shift in meaning that moved the particular sense of 'mass' into a different semantic domain. Furthermore, from the example we see how polysemy drives synonymy, i.e. by making semantic shifts lexical units are pushed into new synonymic relations with the lexical units profiling the same conceptual content in more-or-less the same way. The third sense of 'mass' ('a body of water') illustrates yet another principle by which polysemy structures the Croatian lexicon. Here not only has the semantic shift occured to indicate a specific homogenous and fairly large quantity of water (as in lakes and seas), but its specialization of meaning is further indicated by the collocation *vodena masa* 'lit. water mass, a mass of water'.

Although the example provided was from the category of nouns in CroWN, verbs behave in a similar manner, having even more polysemous senses entering into different synonymous relations and domains (Raffaelli&Katunar, 2010, in press). One notable property of verbs as opposed to nouns is their high degree of schematicity (Fellbaum, 1998), which accounts for a larger number of verb senses as well as smaller number of lexical units pertaining to the category of verbs. For this reason the makers of the original Princeton WordNet describe and categorize semantic verb relations in different terms from nouns, e.g. the relations of troponymy and entailment are considered as verbal counterparts of the noun relations hyperonymy/hyponymy and meronymy, respectively (Fellbaum, 1998). Polysemy of verbs is also described somewhat differently in WordNet. Peters et al. (1998), for instance, distinguish different criteria for sense disambiguation of verbs than that of nouns, such as transitivity/intransitivity, causativity/inchoativity etc. paired with the usage of different syntactic patterns that reflect the semantic shifts of verb lexemes. Miller (1999) and Fellbaum (1998, 2000) also point out repeteadly that polysemy operates under different principles when it comes to verbs as opposed to nouns. However, in the process of building CroWN (Raffaelli et al., 2008, Raffaelli&Katunar, 2010.) we have come face to face with certain regularities in both noun and verb relations that point to more general principles of polysemy working uniformly across categories, such as the ubiquitous mechanisms of metonymy and metaphor motivating the sematic shifts in both categories (see Lakoff, 1987, Langacker, 1987, Raffaelli&Katunar, 2010,

in press). Thus we believe that the aforementioned ways of interaction between synonymy and polysemy illustrated with the noun 'mass' are equally relevant for the same interaction for verbs, as we will show in the rest of the paper.

We will analyze polysemous lexical units in CroWN as defined in their senses by a) the surrouding lexical units of the same synset, b) by the semantic domain and hierarchy to which a particular sense belongs but also c) by specific constructional specifications in the lexical entries (one of these types being the example of 'mass' / 'water mass').

In this paper we deal especially with the last type of interaction mentioned, that of constructional specifications of lexical entries of verbs and we show how it serves to profile and specify the meaning of the category of the verb lexemes. While working on synsets in CroWN, it became apparent that some concepts are, along with one-word units, lexicalized as multi-word units. Though these are mostly mentioned as pertaining to idioms (also discussed in Fellbaum (1998) for English, e.g. 'kick' in *kick the bucket*), we will explore a more direct verbal construction, the V+Prep construction, in detail for the purposes of this paper (e.g. *poslati po* – to call, *zagrijati se za* – to be interested in). It is important to point out that the main verb has a completely different meaning without the preposition, and what is gained by adding a preposition to it is a holistic semantic unit[1] expressing a very different concept (e.g. *zagrijati se* 'warm up', zagrijati se za – to be interested in). We will also show that V+Prep. construction cannot be treated as an idiom, instead, this structure is consistent with what is called phrasal verbs in English (e.g. to make out, to run out). Introducing the concept of phrasal verbs is also very important, because it presents a novelty in the description of Croatian, as well as other Slavic languages. We believe that the incorporation the V+Prep constructions in the description of the Croatian lexicon is thus an important task that not only contributes to the fine-grained analysis of Croatian but also enriches the CroWN database and expands its applications in natural language processing tasks. Along with the incorporation of the V+Prep construction in CroWN, we set out to build a database of Croatian "phrasal verbs". We describe the methods used in building the database and demostrate its applicability to the sense elaboration of verb synsets in CroWN as well as its benefits in the lemmatization of large corpora in the last section of the paper.

2. Phrasal Verbs

Phrasal verbs are a widely accepted phenomenon in languages such as English, and also in Dutch and German (Jackendoff, 2002), but as to our knowledge, there hasn't been a straightforward hypothesis about the existence of phrasal verbs in Slavic languages, including Croatian (cf. Sussex&Cubberley, 2006, Menac, 2007).

Descriptions of phrasal verbs vary from traditional approaches which interpret them as derivationally unpredictible, to cognitive approaches which point out the regularities of their meanings and formation through semantic shifts via metaphor and metonymy (see also Kovács, 2007). Taylor (2002) points out that the link between the verb and the preposition within the phrasal verb structure is notably different than a compositional V+Prep. Thus in the example of 'look up' he shows that the interpretation can be twofold, depending on the compositionality or the bondedness[2] of 'look up':

1. look up the chimney – where 'look' can be replaced by 'peer' or 'gaze' up the chimney, or one can look down the chimney. In other words, the construction is compositional and its components can be replaced;
2. look up a word in the dictionary – where 'look' cannot be replaced by e.g. gaze (*gaze up a word) or any other lexical unit. In other words, „*look* and *up* coalesce to form a semantic unit in which the basic meaning of *up* has been coerced by a metaphorical meaning of *look* (Taylor, 2002: 330).

So, the criteria for identifying a phrasal verb are:

a) the semantic unity of the V+Prep. construction;
b) its distibutional properties which sanction the replacement of any of its parts by any other lexical unit.

Based on Taylor (2002) and other cognitive accounts (Lakoff, 1987, Langacker, 1987, Kovács, 2007 and others) we apply these criteria in the definiton and extraction of Croatian phrasal verbs. To our knowledge, nobody brought attention to the fact that phrasal verbs are not mentioned or described in Croatian. Furthermore, some authors even take the claim: "Phrasal verbs do not exist in Croatian language" (Geld, 2006) as some kind of a starting point in their papers. We find that the reasons for this ommision probably lie in (a) the contrastive analysis of Croatian and English, where prepositions are translationally equated with Croatian prefixes (eng. pull **out** – cro. **iz**vući; Arsenijević, 2004) (b) the fact that Croatian phrasal verbs form a smaller and more restrictive set than in English. However, as we will show in the following section, this set fits in the aforementioned criteria.

For the purposes of this paper two contemporary Croatian grammars[3] and two dictionaries[4] were consulted to see how they are dealing with verb constructions, namely V + Prep. constructions.

When it comes to Croatian grammars, phrasal verbs do not exist as a separate category, moreover, they're not even mentioned as a potential category in Croatian. Grammars that were taken into account mention verb government, but they do not give any detailed description, nor mention how different prepositions influence verb meaning. Government (rection) is simply presented as a verb[5] capacity to require a complement, namely object, in a predefined case. Such a classification is not cleary

[1] What we mean by the "holistic semantic unit" is a unit whose meaning is not simply a sum of its parts, i.e. compositional.

[2] Taylor (2002: 588) defines bondedness as a process „when units combine into a complex expression – especially when the composite form is entrenched and is characterized by coercion – it may be difficult to identify the expression's component units. The units become 'bonded' in a relatively unanalysible structure."

[3] Barić et al. (2003), Silić&Pranjković (2005).

[4] Anić (1991), Šonje (2000).

[5] As well as noun and adjective capacity (Silić&Pranjković, 2005: 263-264).

delimited as to the division between adverbials and object complements, and is sometimes confusing to discern to what it actually refers to. This problem arises from the fact that Croatian grammars do not delimit valency from government, instead they view them as synonymous (Silić&Pranjković, 2005:389) or do not mention valency at all (Barić et al., 2003).[6] As a consenquence of this inadequate description of verb valencies Croatian dictionaries also don't include phrasal verbs, i.e. V+Prep. constructions with shift in meaning as separate lemmas. However, they do recognize a shift in meaning of verbs in different constructions, but list only the main verbs as lexical entries with different senses. Thus, the meaning 'to be interested in' is listed under the lemma *zagrijati se*, but the correlation of shift in meaning and preposition *za* isn't shown. In other words, the user of Croatian dictionaries cannot decode the fact that this particular shift of meaning occurs only in V+Prep. *za* construction.[7] In the only online dictionary of Croatian language[8] the situation is more or less the same, while it is based upon Anić (1991 and later) whose primary purpose was not conceived as a computational resource. It is therefore unhelpful, not only when it comes to individual users but also when it comes to disambiguating senses in machine translation (henceforth MT) systems or even in CroWN.

Thus, it needs to be shown how we can modify the current verb description and lemmatization in Croatian, in order to incorporate the set of phrasal verbs within its framework.

2.1.Semantics of Croatian Phrasal Verbs

In our analysis we were particularly interested in the change of the meaning of the main verb when followed by a particular preposition, in contrast to other prepositions which only function is to introduce several kinds of complements, namely objects or adverbials (see Taylor, 2002). For instance, the verb *zagrijati se* (to warm up) can be followed by different prepositions, among which are *pod* (under), *od* (from) and *za* (for):

1. (a) *zagrijati se pod pokrivačem* (to warm up under the blanket)
   (b) *zagrijati se od trčanja* (to warm up from running)
2. (a) *zagrijati se za lingvistiku* (to be interested in linguistics)
   (b) *zagrijati se za kuhanje* (to be interested in cooking)
   (c) *zagrijati se za Brada Pitta* (lit. to be interested in Brad Pitt; to have the hots for Brad Pitt)
3. *zagrijati se za utakmicu* (to warm up for the game)

It is obvious that in (1 a,b) the prepositions *pod* (under) and *od* (from) are part of the adverbials *pod pokrivačem* (under the blanket) and *od trčanja* (from running). They do not affect the verb's meaning, but only introduce a new circumstance of the action expressed by the main verb (in this particular case the location and the manner, respectively). On the contrary, the preposition *za* (for) in (2), apart from introducing a sentence object, completely changes the meaning of the main verb. *Zagrijati se* (to warm up) is metaphorically reinterpreted in accordance with what we may deem as the conceptual metaphor HAPPY IS WARM – SADNESS IS LACK OF HEAT (Kövesces, 2003), e.g. *ohladiti se od (koga)* (lit. to cool down, to loose interest (in somebody), *izgarati od (ljubavi, želje etc.)* (lit. burn with (love, desire)). Thus, the V+Prep. constuction in (2) expresses a very different concept than the V itself. Although it is clear that the metaphorical shift in meaning has happened and one can state that *to be interested in* is just one of the several meanings of the polysemous verb *zagrijati se*, what we claim is that the preposition is an explicit marker as well as an inherent part of that shift and thus should be a part of a lemma. As the examples in (2 a,b,c) also show, the meaning of the phrasal verb *zagrijati se za* is consistent regardless of the object complement following the preposition (it can be an abstract notion of science, e.g. linguistics or an activity, e.g. cooking or a person of romantic interest, e.g. Brad Pitt). Furthermore, one must be cautious to distinguish the compositional *zagrijati se za* (3) 'warm up' from the phrasal *zagrijati se za* (2 a,b,c) 'to be interested in'. Parallel to Taylor's (2002) description of 'look up' in English, these variants of *zagrijati se za* differ in their meaning in a way that (3) *za* is a part of the PP structure while in (2 a,b,c) is a part of the phrasal verb followed by an object. What follows from this distinction is the necessity to lemmatize *zagrijati se za* in (2a,b,c) 'to be interested in' separately from *zagrijati se* 'warm up'. Even though in (3) we see that *zagrijati se* 'warm up' can take *za* (for) as its complement it does not belong to its lemma because it is substitutable with any preposition and does not affect the verb's meaning. Such a semantic description argues for the separation of monolexemic and phrasal verbs in their lemmatization and notation in CroWN hierarchies.

On the other hand, we need to distinguish such phrasal verb constructions from idioms, i.e. other multi-word units (henceforth MWU). Idioms vary in their components and complexity, whereas phrasal verbs have only the V+Prep structure. Moreover, phrasal verbs illustrate the continuum of linguistic constructions (Fillmore, 1987), falling between the monolexemic verbs and full-fledged idiomatic constructions. Also, phrasal verb meaning is still, as we will demonstrate later, closely related and motivated by the schema of the polysemous structure of the verb itself.

2.2. Croatian Phrasal Verbs Database

Since, as we pointed out, phrasal verbs do not exist as lemmas in Croatian dictionaries, we weren't sure how to include them as literals in CroWN, but keeping them out of CroWN would significantly impoverish our resource.

[6] Conversely, we believe that the correct approach is to define government as referring solely to object complements, i.e. the verb governing the object case. On this account, valency is a broader term than government, and includes all sentence arguments, i.e. both subject, object and adverbial cases. For detailed description of valency in Croatian cf. Šojat (2009).

[7] Only in Šonje (2000) syntagmatic expressions are only vaguely noted in lexical entries as usage examples and not explained further.

[8] Hrvatski jezični portal (Croatian Language Portal), www.hjp.srce.hr. The fact is that HJP is slightly adapted Anić's dictionary.

So the first step we made was to write them down and create a small database of so called Croatian phrasal verbs.

Main verb	Prep.	Case	Synonyms
ciljati	na	ACC. a./i.	misliti
dovesti	do	GEN. i.	uzrokovati
držati	do	GEN. a./i.	cijeniti
ići	na	ACC. i.	poduzeti, namjeravati
ići	za	INST. i.	nastojati, težiti
patiti	od	GEN. i.	bolovati
plivati	u	LOC. i.	snalaziti se
poslati	po	ACC. a.	pozvati
privoljeti	na	ACC. i.	pristati
skinuti se	s	GEN. i.	odviknuti se
tući	po	LOC. a./ i.	pucati
ubiti se	od	GEN. i.	izmoriti se
zagrijati se	za	ACC. a./i.	zainteresirati se
zakačiti se	s	INST. a.	posvađati se
zapaliti se	za	ACC. a./i.	zainteresirati se

Main verb	Prep.	Case	Synonyms
to aim (at)	at	ACC. a./i.	to think
to bring (to)	to	GEN. i.	to cause
to hold	to	GEN. a./i.	to value
to go	on	ACC. i.	to opt for
to go	for	INST. i.	to aim at
to suffer (from)	from	GEN i.	to be ill, to suffer from
to swim (into)	into	LOC. i.	to get along
to send (for)	over	ACC. a.	to call
to persuade (to)	on	ACC. i.	to accept
to take (off)	with	GEN. i.	to quit
to beat	over	LOC. a./i.	to shoot
to kill (oneself)	from	GEN. i.	to exhaust oneself
to warm (up)	for	ACC. a./i.	to be interested in
to attach (to)	with	INST. a.	to fall out with
to burn (up)	for	ACC. a./i.	to be interested in

Figure 1 Sample of the Croatian phrasal verb database followed by an English translation

Our database includes the following data:
1. main verb – lemma in current dictionaries of Croatian language;
2. preposition – only the particular preposition which changes the meaning of the main verb in a specific way is listed;
3. case of the complement following the preposition along with its animacity (a.)/inanimacity(i.);
4. synonym(s). (for the sample see. Figure 1 below)
For example:
zagrijati se za A (a.)/(i.) zainteresirati se, zanijeti se
V Prep. case synonyms
'to be interested in'.
Since there is no such thing as a lexicon or dictionary of Croatian verbs including prepositions following them, we weren't able to automatically extract all V+Prep. constructions, in order to find possible candidates for Croatian phrasal verbs database. Thus the manual making of the database is also a prerequisite for automatic extraction of phrasal verbs from corpora. Since our primary goal is to enrich CroWN with phrasal verbs we started out by manually examining the list of about 2 300 verb synsets currently present in CroWN and extracting possible candidates for phrasal verbs. Those were primarily verbs with several senses whose synonyms in the same synset were indicative of a semantic shift occuring in the phrasal verb candidate. For instance, *ciljati* 'to aim at' appears in two synsets, one being defined as 'the act of aiming a weapon at somebody/something' and its synonym being the verb *nišaniti* 'to aim a weapon at'; the other synset contains the units *ciljati (na)* but also *misliti* 'to think', clearly indicating that *ciljati (na):2* has undergone a semantic shift into the domain of cognition and is also followed by the particular preposition, in this case *na* 'on'. So the second sense of *ciljati na* was treated as a phrasal verb candidate. The candidates extracted from the list of verbs in CroWN were then cross-referenced with their occurences in the CNC[9] in order to establish their syntactic patterns and distribution, i.e. to check whether they satisfy the two criteria for defining phrasal verbs (as listed above), the semantic unity of the MWU and its distribution. Its distributional pattern, i.e. the case occuring with a particular phrasal verb was also added to the database.[10] Furthermore, we started to develop a lexicon of Croatian verbs containing their derivational and inflexional forms, as well as their valency frames. This will facilitate detection of an even greater number of phrasal verb candidates in two ways:
1. when construing verb valency frames[11], we could recognize V+Prep. constructions which form holistic semantic units and include them in our database;
2. after construing verb valency frames, we could more easily extract all V+Prep. constructions in order to detect phrasal verbs among them.
This will be an important step towards expanding the database, since we have managed to manually extract 76 candidates so far, which may seem as a small sample, but it still comprises 3,2% of the current CroWN verb synsets and is highly indicative of a more widespread phenomenon in the Croatian lexicon.
The database will then be used to incorporate all detected phrasal verbs into CroWN, more precisely into synsets

[9] Croatian National Corpora, www.hnk.ffzg.hr.
[10] Another important aim is to get a general list of prepositions that can stand as a prepositional part of phrasal verbs in Croatian language. So far eight prepositions are extracted in our database, among them *za* (for) and *na* (on) being most frequent. The current list of prepositions could help us to extract more phrasal verbs from CNC by listing V+Prep constructions in more narrow way - we don't have to include all prepositions in Croatian, but only those that appear in the existing database.
[11] Construing verb valency frames in Croatian is almost completely manual work. There is only one printed Croatian valency dictionary *Rječnik valentnosti hrvatskih glagola* (Croatian Valency Dictionary), which is restricted to a very small set of verbs and does not give a complete description of valency frames, especially when it comes to the prepositions required by the verb. Much larger in size and quantity is *Crovallex* (Mikelić Preradović et al., 2009), an electronic lexicon of Croatian verbs which resembles in its structure to Czech lexicon *Vallex* (Žabokrtský, Lopatková, 2007). See Šojat (2009).

which contain synonyms listed next to the them in the database. This implies that we would treat phrasal verbs as a separate lemmas in CroWN which would also have different synonyms, hyperonyms etc. than the main verb of the phrasal construction. It also means that once the list of phrasal verbs is complete and added to CroWN we could simply add the list to the list of lemmas in CNC and thus lemmatize the entire corpus. In the next chapter we illustrate the interaction of polysemy and synonymy as reflected in the CroWN structure pertaining to phrasal verbs and their semantic relations.

2.3. Semantic Relations of Phrasal Verbs in CroWN

There are two important aspects of the interaction of synonymy and polysemy with regards to phrasal verbs. First, phrasal verbs are specifications of the more schematic meaning denoted by the main verb via prepositions. Secondly, since polysemy drives synonymy, these verbs are also placed in different synsets as well as different lexical hierarchies, which implies a whole new set of semantic relations gained by the semantic shift in specialization. On the other hand, their relation to the other senses of the main verb in CroWN is preserved through the inclusion of the sense of a phrasal verb as one of the senses of its main verb. To illustrate this point, we will describe and show the semantic relations of the verb *držati* 'to hold'. Since this is a highly polysemous verb in Croatian, it has a plentitude of senses registered in CroWN, one of them being specified by a phrasal verb *držati do* 'to value'. All together, the verb *držati* 'to hold' has 13 senses, the thirteenth being the sense *držati do* 'to value'. Below in Figure 2 is the entire synset to which it belongs, along with its synonym pairs, definition and usage examples (followed by its PWN counterpart).

<SYNSET>
<ID>ENG20-00670967-v</ID>
<POS>v</POS>
<SYNONYM>
<LITERAL>**cijeniti**<SENSE>2</SENSE></LITERAL>
<LITERAL>**štovati – poštovati - poštivati**<SENSE>2</SENSE></LITERAL>
<LITERAL>**držati do**<SENSE>13</SENSE></LITERAL>
<LITERAL>**respektirati**<SENSE>9</SENSE></LITERAL>
</SYNONYM>
<DEF>imati visoko mišljenje o komu ili čemu; uvažavati čije mišljenje</DEF>
<USAGE>Visoko cijenim njezino sposobnosti.</USAGE>
<USAGE>Poštujem tvoju slobodu govora.</USAGE>
<USAGE>Držim do tvojeg mišljenja.</USAGE>
<BCS>2</BCS>
<DOMAIN>factotum</DOMAIN>
<SUMO>IntentionalPsychologicalProcess<TYPE>+</TYPE></SUMO>
<CROWN>1</CROWN>
</SYNSET>

<SYNSET>
<ID>ENG20-00670967-v</ID>
<POS>v</POS>
<SYNONYM>
<LITERAL>**respect**<SENSE>1</SENSE></LITERAL>
<LITERAL>**esteem**<SENSE>1</SENSE></LITERAL>
<LITERAL>**value**<SENSE>3</SENSE></LITERAL>
<LITERAL>**prize**<SENSE>3</SENSE></LITERAL>
<LITERAL>**prise**<SENSE>3</SENSE></LITERAL>
</SYNONYM>
<DEF>regard highly; think much of </DEF>
<USAGE>I respect his judgement.</USAGE>
<USAGE>We prize his creativity.</USAGE>
<BCS>2</BCS>
<DOMAIN>factotum</DOMAIN>
<SUMO>
IntentionalPsychologicalProcess<TYPE>+</TYPE></SUMO>
</SYNSET>

Figure 2 Phrasal verbs in CroWN synsets

In the example it is clear that *držati do* 'to value' enters into rather different synonymic relations than for instance *držati* 'to hold (in ones hand)'. As the synonyms surrounding *držati do* 'to value' indicate, the meaning of the phrasal verb *držati + do* 'lit. hold to, to value' is far removed from the domain of physically grasping on object (as in 'to hold in ones hand') and pertains to the domain of psychological processes, namely those including respect and judgement. The semantic shift here is cleary metaphorical, as it includes a movement from a concrete domain (physical object interaction of 'holding') to the abstract domain of judgement. The connotations added to the abstract notion of 'holding to or valuing' are further motivated by the domain of judgement. Thus we see that the polysemous shift motivated the verb to specialize in meaning and enter synonymic relations with 'respect' and 'value', which otherwise would not be possible. Futhermore, it is important to stress that the monolexemic verb *držati* 'to hold' would not be able to enter these relations because it would not have been specified enough as to its meaning, i.e. the only possibilty is to have a phrasal verb as lemma in CroWN since the option of entering only the main verb would leave the relations in this particular synset understated and vague.

To further stress the importance of proper specification of lemmas and their polysemous relations in CroWN, we will present the entire polysemous structure of the polysemous verb *držati* 'to hold', taken and modified from Raffaelli&Katunar (2010, in press). Raffaelli&Katunar (2010, in press) do not include in their analysis phrasal verbs and do not treat them as separate lemmas in CroWN, although they discuss in detail the ways of presenting polysemous verbs as radial structures. Thus we modify the existing graph (see Figure 3. below) of *držati* 'to hold' in order to show how the inclusion of phrasal verbs adds relevant information about parts of the radial structure containing phrasal verbs as well as the structure of the Croatian lexicon.

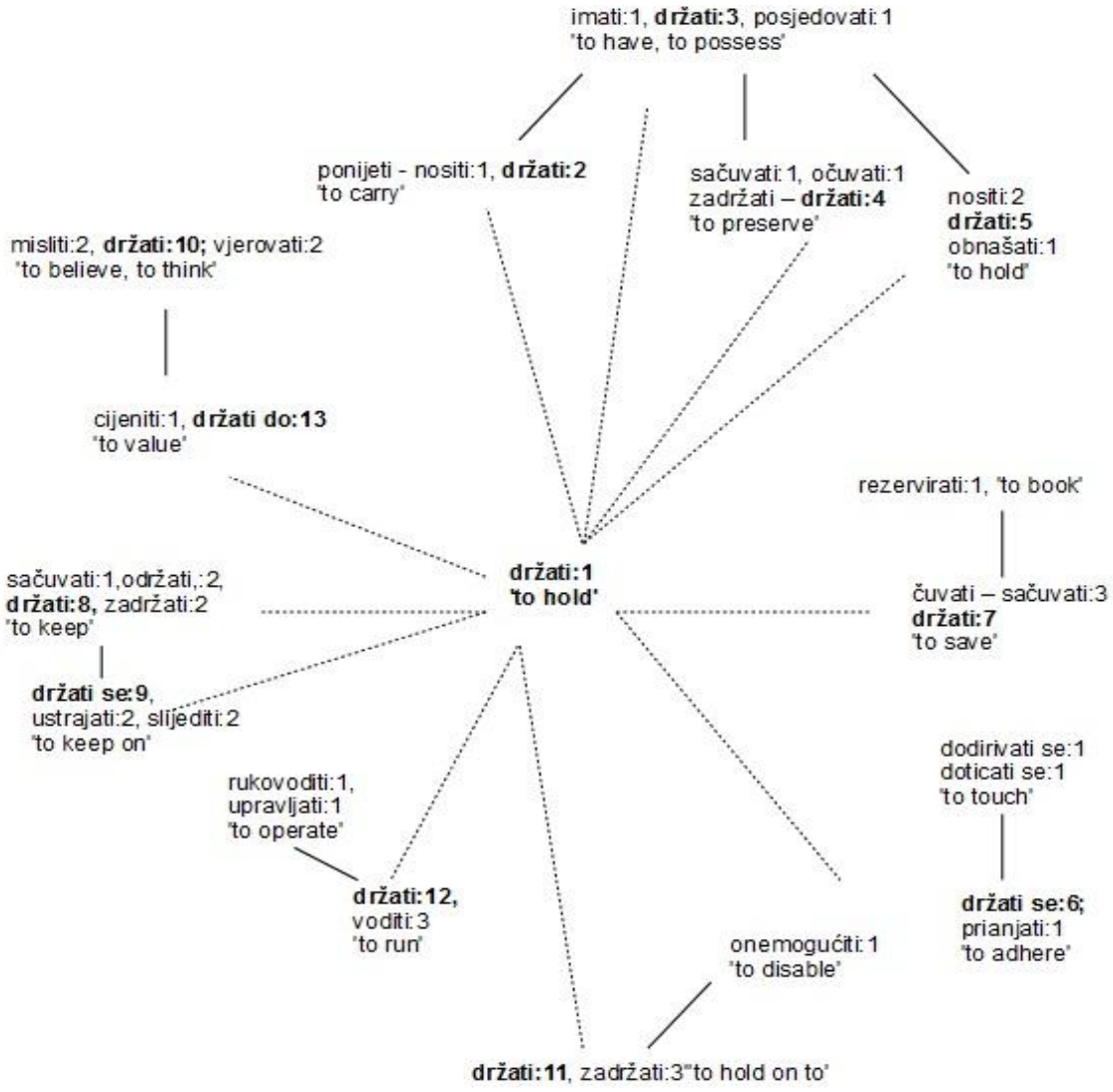


Figure 3 Semantic relations of the verb *držati* 'to hold' and its senses in CroWN. Above each sense are the hyperonymic synsets noted by the continous lines. The dotted lines represent sense extensions from the source meaning *držati:1* 'to hold physically in one's hands'.

The polysemy of *držati* 'to hold' is very clearly shown in Figure 3., where the verb has 13 senses that vary from the concrete sense of 'holding in ones hands' to the senses of 'keeping', 'thinking', 'possesion', 'adhering' etc. What Figure 3. also indicates is the path of the semantic shift from the source meaning *držati* 'to hold' to *držati do* 'to value'. The shift is not a direct one, but it includes a) the metaphorical shift from *držati:1* 'to hold' to *držati:10* 'to think, to believe' motivated by the fact that one can 'hold an opinion or belief' in the abstract sense, and b) the specialization of *držati:10* 'to think, to believe' by the features of judgement and esteem added by the preposition *do* 'to' in *držati do:13* 'to value'. In other words, the link between *držati:10* 'to think' and the more specific *držati do* 'to value' is best described in the way that *držati do* 'to value' specifies a particular manner of thinking, that of 'holding on to' a person, opinion etc., which implies the relevancy of the entity one is 'holding on to' or 'thinking of', allowing it to have a value component of its meaning.

It is clear from the example in Figure 3. that by adding the V+Prep construction we describe the properties of the entire radial structure in more detail, and represent the semantic shifts, especially specification in this case, as processes transparently noted in the lemmas themselves, i.e. in the preposition added to the main verb. What this allows is an expansion of synonymic and polysemous relations in CroWN, as well as (in some cases) the inclusion of phrasal verb into new hierarchical relations with which they otherwise had no relation at all as monolexemic units (see example above *zagrijati se za* 'to be interested in').

3. Conclusion and Future Work

In this paper we presented the description of phrasal verbs in Croatian, which to our knowledge are ommitted from

any current and past descriptions of the Croatian lexicon and grammar. We emphasized the importance of this description from the viewpoint of a) a fine-grained analysis of semantic relations in CroWN, and b) the interaction of synonymy and polysemy as manifested in the semantic relations of phrasal verbs to their monolexemic counterparts and facilitated by the structure of CroWN, and c) current lemmatization of phrasal verbs in Croatian dictionaries and its modification for the necessities of CroWN. For these purposes we proposed building a database of Croatian phrasal verbs, described its structure and the methods of its further expansion. Future work includes building valency frames which would enable this expansion, but also the extraction of V+Prep constructions in large corpora and incorporation of the extracted phrasal verbs into CroWN verb hierarchies. We believe that this work will contribute to (a) the theoretical aspects of the interaction between polysemy and synonymy; (b) description of the Croatian verb system; (c) the enrichment of semantic relations in CroWN; (d) lemmatization of verbs in CroWN and other resources such as CNC; (e) facilitating MT applications and information extraction via CroWN.

4. References

Anić, V. (1996). *Rječnik hrvatskoga jezika* [The Dictionary of Croatian Language]. Zagreb: Novi Liber.

Arsenijević, B. (2004). *Non-predicative Particles as Adjuncts to Abstract Arguments*. An abstract from the CHRONOS 6 Conference in Geneva, 2004. http://www.unige.ch/lettres/latl/chronos/Arsenievic.pdf

Barić, E. et al. (2003). *Hrvatska gramatika* [Croatian Grammar]. Zagreb: Školska knjiga.

Dehé, N., Jackendoff, R. et al. (eds.) (2002). *Verb-Particle Explorations (= Interface Explorations 1).* Berlin / New York: Mouton de Gruyter.

Geld, R. (2006). Strateško konstruiranje značenja engleskih fraznih glagola [Strategic Construal: English Particle Verbs]. *Jezikoslovlje*, 7(1-2), pp. 67 – 111.

Fellbaum, Ch. (ed.) (1998). *WordNet. An Electronic Lexical Database. With a preface by George Miller.* Cambridge, MA: MIT Press.

Fellbaum, Ch. (1998). Towards a Representation of Idioms in WordNet. In *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*. Montreal: COLING/ACL, pp. 52 – 57.

Fellbaum, Ch. (2000). Autotroponymy. In Ravin, Y. and Leacock, C. (Eds.), *Polysemy*, pp. 52 – 67. Cambridge: Cambridge University Press.

Fillmore, Ch. and Atkins, B.T. (2000). Describing Polysemy: the Case of Crawl. In Ravin, Y. and Leacock, C. (Eds.), *Polysemy*, pp. 91 – 110. Cambridge: Cambridge University Press.

Fillmore, Ch., Kay, P., O'Connor, M. K. (1988). Regularity and Idiomaticity in Grammatical Constructions: The Case of *let alone*. *Language* 64, pp. 501 – 538.

Kovács, É. (2007). The Traditional vs. Cognitive Approach to English Phrasal Verbs. *English.:* http://epa.oszk.hu/02100/02137/00022/pdf/EPA02137_ISSN_1219-543X_tomus_16_fas_1_2011_141-160.pdf

Kövecses, Z. (2000). *Metaphor and Emotion: Language, Culture, and Body in Human Feeling*. Cambridge: Cambridge University Press.

Lakoff, G.. (1987). *Women, fire and dangerous things: What Categories Reveal about the Mind.* Chicago/London: The University of Chicago Press.

Langacker, R.W. (1987). *Foundations of Cognitive Grammar. Volume 1. Theoretical Prerequisites.* Stanford: Stanford University Press.

Menac, A. (2007). *Hrvatska frazeologija* [Croatian Phraseology]. Zagreb: Knjigra.

Mikelić Preradović, N., Boras, D., Kišiček, S. (2009). CROVALLEX: Croatian Verb Valence Lexicon. In *Proceedings of the ITI 2009 31st International Conference of Information Technology Interfaces.* Zagreb: SRCE, pp. 533 – 538.

Miller, G.A. (1999). Nouns in WordNet. Fellbaum, C. (Ed.), *WordNet. An Electronic Lexical Database.* Cambridge/Massachussetts/London : MIT Press, pp. 23 – 47.

Peters, W. et al. (1998). Cross-linguistic Alignment of Wordnets with an Inter-Lingual_Index. Vossen, P. (Ed.), *EuroWordNet: A multilingual database with lexical semantic networks.* Dordrecht/Boston/London: Kluwer Academic Publisher, pp. 221 – 225.

Raffaelli, I. et al. (2008). Building Croatian WordNet. In Tanács, A. et al. (Ed.) *Proceedings of the 4th Global WordNet Conference.* Szeged: Global WordNet Association.

Raffaelli, I., Katunar, D. (2010). Leksičko-semantičke strukture u Hrvatskom WordNetu [Lexical-semantic structures in the Croatian WordNet]. *Filologija* (in press).

Silić, J., Pranjković, I. (2005). *Gramatika hrvatskoga jezika za gimnazije i visoka učilišta* [Croatian Grammar for High Schools and Universities]. Zagreb: Školska knjiga.

Sussex, R., Cubberley, P. (2006). *The Slavic languages. (Cambridge Language Surveys).* Cambridge: Cambridge University Press.

Šojat, K. (2009). Morfosintaktički razredi dopuna u Hrvatskom WordNetu [Morphosyntactic Annotation in the Croatian WordNet]. *Suvremena lingvistika* 68, pp. 305 – 339.

Šonje, Jure (Ed.) (2000). *Rječnik hrvatskoga jezika* [The Dictionary of Croatian Language]. Zagreb: Leksikografski zavod Miroslav Krleža: Školska knjiga.

Taylor, J. R. (2002). *Cognitive Grammar.* New York : OUP.

Žabokrtský, Z., Lopatková, M. (2007). Valency Information in VALLEX 2.0.: Logical Structure of the Lexicon. *The Prague Bulletin in Mathematical Linguistics*, 87, pp. 41 – 60.

Designing a Database of GermaNet-based Semantic Relation Pairs involving Coherent Mini-Networks

Silke Scheible and Sabine Schulte im Walde

Institute for Natural Language Processing (IMS)
University of Stuttgart, Germany
scheible@ims.uni-stuttgart.de, schulte@ims.uni-stuttgart.de

Abstract

We describe the design and compilation of a new database containing German semantic relation pairs drawn from the lexical network GermaNet. The database consists of two parts: A representative selection of lexical units drawn from the three major word classes adjectives, nouns, and verbs, which are balanced according to semantic category, polysemy, and type frequency ('SemrelTargets'); and a set of semantically coherent GermaNet subnets consisting of semantic relations pairs clustering around the selected targets ('SemrelNets'). The database, which contains 99 SemrelTargets for each of the three word classes, and a total of 1623 relation pairs distributed across the respective subnets, promises to be an important resource not only for research in computational linguistics, but also for studies in theoretical linguistics and psycholinguistics. Currently, the data is being used in two types of human judgement experiments, one focusing on the generation of semantically related word pairs, and the other on rating the strength of semantic relations.

1. Introduction

Paradigmatic semantic relations such as synonymy, antonymy, hypernymy/hyponymy, and co-hyponymy have been the focus of many studies in theoretical and applied linguistics (Cruse (1986); Lyons (1977); Murphy (2003)). Approaches in computational linguistics also addressed paradigmatic relations, especially synonymy (e.g., Edmonds and Hirst (2002); Curran (2003); Lin et al. (2001)) and hypernymy (e.g., Hearst (1992); Caraballo (2001); Snow et al. (2004)), but less so antonymy, and often with respect to modelling contradiction (e.g., Lucerto et al. (2004); Harabagiu et al. (2006); de Marneffe et al. (2008)). Many approaches included one or the other paradigmatic relation within a set of target relations (e.g., Pantel and Pennacchiotti (2006); Morris and Hirst (2004); Turney (2006)), but to our knowledge no earlier work has specifically focused on all standard paradigmatic relations.

Over the years a number of datasets have been made available for studying and evaluating semantic relatedness. For English, Rubenstein and Goodenough (1965) obtained similarity judgements from 51 subjects on 65 noun pairs, a seminal study which was later replicated by Miller and Charles (1991), and Resnik (1995). In 2001, Finkelstein et al. (2002) created a set of 353 English noun-noun pairs rated by 16 subjects according to their semantic relatedness on a scale from 0 to 10. For German, Gurevych (2005) replicated Rubenstein and Goodenough's experiments by translating the original 65 word pairs into German. In later work, she used the same experimental setup to increase the number of word pairs to 350 (Gurevych, 2006).

Zesch and Gurevych (2006) note a number of shortcomings of previous approaches to creating datasets of semantic relatedness. First of all, they state that manually compiled lists of word pairs are often biased towards highly related pairs. They further draw attention to the fact that previous studies considered semantic relatedness of *words* rather than *concepts*, noting that polysemous or homonymous words should be annotated on the level of concepts. To overcome these limits for German, they propose automatic corpus-based methods which they employ to create a set of 328 related concept pairs across different word classes and drawn from three different domain-specific corpora. While this approach enables fast development of a large domain-specific dataset covering all types of lexical and semantic relations, they found that highly related concept pairs were under-represented in their data.

In this paper we describe the design and compilation of a new large-scale dataset containing German concept pairs related via paradigmatic semantic relations, which is currently being annotated with human judgements on the relations. Like Zesch and Gurevych (2006), our approach involves automatic compilation methods and a focus on concepts rather than words. However, in contrast to their approach, our data is drawn from GermaNet (Lemnitzer and Kunze, 2007), a broad-coverage lexical-semantic net for German, using a principled sampling technique. The resulting dataset consists of two parts:

1. A representative selection of lexical units drawn from the three major word classes adjectives, nouns, and verbs, which are balanced according to semantic category, polysemy, and type frequency (referred to as 'SemrelTargets'); and

2. A set of salient semantic GermaNet subnets consisting of paradigmatic semantic relations clustering around each of these targets ('SemrelNets').

The semantically coherent subnets (illustrated in Figure 1) allow an assessment of concepts within their semantic neighbourhood, and the stratified sampling technique ensures that the dataset contains a broad variety of relation pairs. The data is currently being used in two types of human judgement experiments: One focusing on the generation of semantically related word pairs, and the other on human rating of the strength of semantic relations.

The dataset contains a set of target lexical units (99 SemrelTargets each for the three word classes) and 1623 relation

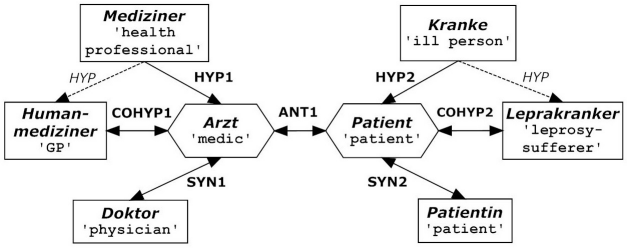


Figure 1: Example of a SemrelNet (Target *Arzt*, 'doctor')

pairs distributed across the respective subnets, thus representing one of the largest principled datasets for studying semantic relations. We anticipate that it will not only be of considerable use in computational areas in which semantic relations play a role (such as Distributional Semantics, Natural Language Understanding/Generation, and Opinion Mining), but also in studies in theoretical linguistics and psycholinguistics.

This paper introduces the selection criteria and tools which were implemented to extract the set of SemrelTargets and their associated SemrelNets from GermaNet[1]. Section 2 aims to provide further motivation for the creation of this dataset by giving a brief overview of the research project it is part of, and discussing potential applications of this work. After introducing the database GermaNet, from which our data was sampled (Section 3), we describe the sampling procedure employed to select the set of target lexical units (Section 4). Section 5 deals with the notion of 'SemrelNets', and provides a detailed overview of the algorithm and associated tool for building these networks. Finally, in Section 6 we outline two human judgement experiments that are currently in progress, which are based on the dataset described in this paper.

2. Motivation

The compilation of the semantic relations dataset is part of a larger research project in the area of distributional semantics. One major goal of this project is to enhance computational work on paradigmatic semantic relations such as synonymy, antonymy, hypernymy, hyponymy, and co-hyponymy. While paradigmatic relations have been extensively researched in theoretical linguistics and psycholinguistics, they are still notoriously difficult to identify and distinguish computationally, because their distributions in text tend to be very similar. For example, in the sentence 'The boy/girl/person loves/hates his cat', the co-hyponyms *boy*, *girl*, and *person* as well as the antonyms *love* and *hate* occur in identical contexts. We are particularly interested in a theoretically and cognitively adequate selection of features to model word relatedness, paying special attention to word senses and any resulting ambiguities, an issue which is a well-known problem in computational linguistics in general, but which has been largely disregarded in distributional semantics.

[1]Both data and tools will be made freely available on our project homepage (http://www.ims.uni-stuttgart.de/projekte/semrel/resources.html).

In order to address these goals we require a sufficiently large amount of human-labelled data, which may both serve as seeds for a computational approach, and provide a gold-standard for evaluating the resulting computational models. In particular, we plan to make use of two types of human-generated data: (1) Human suggestions of semantically related word pairs, and (2) Human ratings of semantic relations between word pairs. The dataset described in this paper has been designed to enable these studies, and Section 6 will provide further details on the human judgement experiments carried out on the basis of this data.

While the dataset was designed with specific goals in mind, its general design and the associated extraction tools will also be of interest for other areas of NLP and linguistic research, for example Opinion Mining and Sentiment Analysis (where it is important to be aware of synonymy/hypernymy vs. antonymy in order to keep track of continuing vs. changing opinions/sentiments); Statistical Machine Translation (where it is important to be aware of the semantic relations between words because this can help in translation); and Word Sense Disambiguation (where the networks should be able to help with sense definitions in the Gold Standards). In addition, our dataset will also be of major interest for research groups working on automatic measures of semantic relatedness, as it allows a principled evaluation of such tools.

Finally, since our data is drawn from the GermaNet database, our results will be directly relevant for assessing, developing, and maintaining this resource. The random selection of SemrelTargets balanced by semantic category, number of senses and corpus frequency allows a systematic assessment of any biases in the semantic taxonomy. Coupled with further analyses, the evaluation can be as deep as the developer wants it to be. For example, we are currently analysing the random choices with respect to morphological properties, such as simplex vs. complex, and more specifically the types of noun compounds and particle verbs, etc. In the same vein, the SemrelNets point the developer to semantic areas that are particularly (non-) dense. Differences between densities in the networks are expected, they have been shown to be problematic in lexical hierarchies of this kind (Jiang and Conrath, 1997). The SemrelNets allow developers to systematically check if a very low/strong density is appropriate for a specific subnetwork, or if the network is under-/over-represented at that point.

3. GermaNet

GermaNet is a lexical-semantic word net that aims to relate German nouns, verbs, and adjectives semantically. GermaNet has been modelled on Princeton WordNet for English (Miller et al. (1990); Fellbaum (1998)) and shares its general design principles (Kunze and Wagner (1999); Lemnitzer and Kunze (2007)). For example, lexical units denoting the same concept are grouped into synonym sets (so-called 'synsets'). These are in turn interlinked via conceptual-semantic relations (such as hypernymy) and lexical relations (such as antonymy). For each of the major word classes, the databases further take a number of semantic categories into consideration, expressed via top-level

Senses	Freq	Gefühl	Verhalten
1	low	-	*satanisch*, 'satanic'; *gesprächsbereit* , 'ready to talk'
	mid	*empört*, 'indignant'	*naiv*, 'naive'; *schützend*, 'protective'
	high	*witzig*, 'funny'	*rassistisch*, 'racist'; *geschickt*, 'adept'
2	low	-	*drollig*, 'cute'
	mid	*reichhaltig*, 'rich'	*unruhig*, 'unsettled'
	high	*düster*, 'gloomy'	*unschuldig*, 'innocent'
3	low	*furios*, 'furious'	*erledigt*, 'done'
	mid	*heiter*, 'cheerful'	*faul*, 'lazy'; *energisch*, 'energetic'
	high	*wild*, 'wild'	*locker*, 'casual'; *mild*, 'mild'

Table 1: Selection of adjectival SemrelTargets for the semantic categories "Gefühl" ('feeling') and "Verhalten" ('behaviour') in GermaNet

nodes in the semantic network (such as 'Artefakt/artifact', 'Geschehen/event', or 'Gefühl/feeling'). However, in contrast to WordNet, GermaNet also includes so-called 'artificial concepts' to fill lexical gaps and thus enhance network connectivity, and to avoid unsuitable co-hyponomy (e.g. by providing missing hypernyms or hyponyms). GermaNet also differs from WordNet in the way in which it handles part of speech. For example, while WordNet employs a clustering approach to structuring adjectives, GermaNet uses a hierarchical structure similar to the one employed for the noun and verb hierarchies. Finally, the latest releases of WordNet and GermaNet also differ in size: While WordNet 3.0 contains at total of 117,659 synsets and 155,287 lexical units, the respective numbers for GermaNet 6.0 are considerably lower, with 69,594 synsets and 93,407 lexical units.

As GermaNet encodes all types of relation that are of interest for our project (synonymy, antonymy, hypernymy, and co-hyponymy)[2], we decided to choose it as primary source for our data sets. However, it is important to be aware of the fact that GermaNet is largely based on manually compiled sources such as thesauri, which tend to list the most salient semantic relations between words. This means that the inclusion of an entry often depends on the subjective decision of the lexicographer. Nevertheless, GermaNet is still the largest database of its kind for German, and we therefore decided to use it as starting point for our dataset.

4. Dataset I: SemrelTargets

4.1. Design

The purpose of collecting Dataset I was to acquire a broad range of lexical items which could be used as targets in generating semantically related word pairs on the one hand (cf. Section 6), and as targets for the automatic extraction of SemrelNets on the other, to create a coherent set of semantic relation pairs (to be described in Section 5). The targets were sampled randomly from GermaNet following four selection criteria. Three of these criteria were based on information available in GermaNet (part of speech, semantic category, and number of senses). A fourth criterion, corpus frequency, was established externally, since (unlike in WordNet) frequency information is not available in GermaNet. Also, we preferred to rely on larger corpus resources for frequency estimation. With no sense-tagged corpus available for German, we acquired type frequencies from a large lemmatised corpus of German (sdeWaC-3[3]). This means that lexical units (corresponding to word senses) were sampled from GermaNet according to the frequency of the corresponding lemma, and not according to the frequency of the sense itself. For polysemous targets, the frequency provided therefore subsumes the target's associated senses and semantic categories.

We used a stratified sampling procedure where for each of the three parts of speech *adjective*, *noun*, and *verb*, 99 targets were sampled randomly (but proportionally) from the following groups:

1. **Semantic categories:** *16* for adjectives, *23* for both nouns and verbs
2. **Three polysemy classes:** I) *Monosemous*, II) *Two* senses, and III) *More than two* senses
3. **Three frequency classes[4]:** I) *Low* (200–2,999), II) *Mid* (3,000–9,999), and III) *High* ($\geq$10,000)

Initially, for each part of speech, the number of lexical units required from each semantic category was established (proportionally to the total number of lexical units in the respective category), which in turn were distributed proportionally across the three polysemy classes and the three frequency classes[5]. Lexical units matching these criteria were then drawn randomly from GermaNet to populate the data set.

4.2. Results and discussion

Table 1 illustrates the choice of adjectives from the semantic categories "Gefühl" ('feeling') and "Verhalten" ('behaviour'). The former contains 7.38% of all adjectives in GermaNet (633 out of 8582). Correspondingly, a total of 7 adjectives (7.38% of 99) was drawn from this category to be included in the SemrelTargets dataset, and distributed proportionally across the nine sense and frequency classes. Similarly, the category "Verhalten" contains 13.76% of all adjectives (1181 out of 8582), from which 14 were sampled for our dataset, shown in the column labelled 'Verhalten'.

[2]GermaNet 6.0 contains a total of 74,945 hypernymy relations, and 1,587 antonymy relations.

[3]http://wacky.sslmit.unibo.it/doku.php?id=corpora

[4]Type frequency in sdeWaC-3 (Total size: 0.88 billion words)

[5]The class membership thresholds for polysemy and frequency were set manually.

Table 2 shows the distribution of polysemy in the dataset. Since polysemy classes I and II are defined to contain lexical units with exactly 1 and 2 senses, respectively, one third (33) of all selected targets for each word class are monosemous, and another third (33) have two senses. Table 2 shows the number of senses of the remaining 33 lexical units randomly sampled for each word class. The results indicate that the number of lexical units in GermaNet rapidly decreases for sense inventories greater than 3.

Senses	Adj	N	V
1	33	33	33
2	33	33	33
3	29	24	14
4	1	6	7
5	0	1	5
6	1	1	3
7	1	1	2
8	0	0	0
9	0	0	2
10	1	0	0

Table 2: Number of senses selected for each word class

Finally, Figure 2 shows that the sampled data conforms to commonly assumed models of sense frequency distributions: The more senses a lexical unit has, the larger its type frequency in corpus data. Thus, the average frequency of lexical units with one sense is 10,255, while the frequency values of lexical units with two senses and three or more senses are 18,257 and 37,479, respectively.

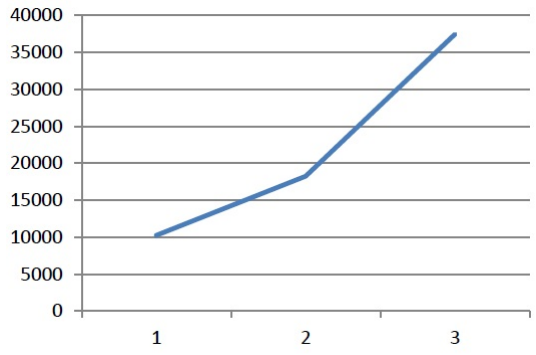


Figure 2: Average frequency per sense class (1=Monosemous, 2=Two senses, 3=More than two senses)

5. Dataset II: SemrelNets

5.1. Design

The targets generated in the previous section were used to build a second dataset containing semantically related word pairs drawn from GermaNet. The goal was to include examples of the following four major types of paradigmatic semantic relations:

1. Antonymy (ANT)
2. Synonymy (SYN)
3. Hypernymy (HYP)
4. Co-Hyponymy (COHYP)

Instead of drawing random relations from GermaNet for each of the input targets, a more sophisticated approach was taken: For each input target, a semantically coherent 'mini-network' of semantic relations was constructed using the target lexical unit (referred to as *t*) as starting point. These interconnected 'SemrelNets' aim to capture a sample of the semantic neighbourhood of *t* (in terms of synonymy, hypernymy, and co-hyponymy), as well as of its opposing one, that is, the neighbourhood of a concept that is opposite in meaning to *t*. In practice, this means that a SemrelNet N typically has the following characteristics:

- N contains a maximum of eight relations (two instances of each type): {ANT1, ANT2, SYN1, SYN2, HYP1, HYP2, COHYP1, COHYP2}.
- N contains two subnets $\{N_1, N_2\}$, where N_1 clusters around the node containing the target lexical unit *t*, while N_2 clusters around a lexical unit which stands in an antonym relation (ANT1) to *t*.
- N_1 typically contains {SYN1, HYP1, COHYP1}, while N_2 contains {SYN2, HYP2, COHYP2}.

A schematic representation of a SemrelNet N is shown in Figure 3. In this example, the boxes labelled *t* and *a1* represent the core nodes of N, and are related via antonymy (ANT1). A second antonymy link (ANT2) is chosen such that it links the synonym of *t* (i.e., *s1* as SYN1) and the synonym of *a1* (i.e., *s2* as SYN2). The antonym-synonym configuration is completed by a hypernym and a co-hyponym of the core nodes *t* and *a1*. Figure 4 shows an actual example from our data illustrating the type of SemrelNet schematised in Figure 3.

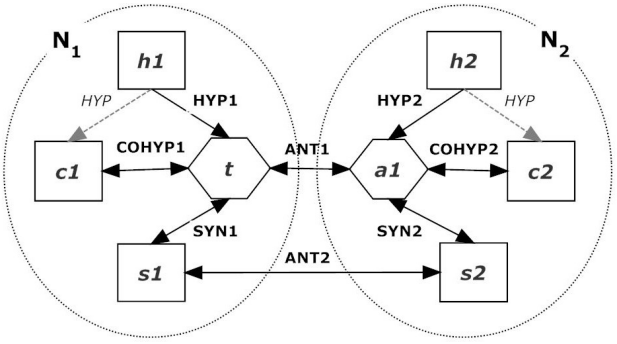


Figure 3: Schematic representation of a SemrelNet

We designed an algorithm for building SemrelNets from target lexical units in GermaNet, of which we provide an overview in the following paragraphs. One important consideration in designing the nets was to find an appropriate balance between network density and random choice of members. For our purposes, SemRelNets with higher density (i.e. with a small number of highly connected nodes) are preferable to more open networks with a larger number of nodes, as the former allows a more principled investigation of the semantic relations of specific lexical items (in particular, the input target), and their perception. For

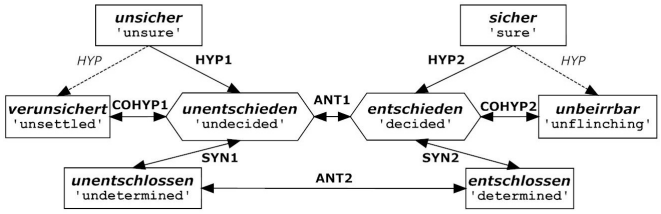


Figure 4: SemrelNet for target *unentschieden* ('undecided')

example, we assume that some paradigmatic semantic relations are more easily distinguished or confused than others, e.g., synonymy is assumed to be easily confused with hypernymy, while antonymy is assumed to be easily confused with co-hyponymy. This is to be confirmed by our experiments. On the other hand, the choice of related nodes should be as random as possible to avoid a bias towards selecting highly connected nodes in GermaNet (which may represent lexical units of higher frequency and/or higher prominence in the lexicographer's mental lexicon). The algorithm tries to take this into account by employing four main methods of locating suitable relation nodes in GermaNet, ordered here according to priority:

- Method 1: Direct-motivated
- Method 2: Direct-random
- Method 3: Indirect-random
- Method 4: Broken-random

In the schematic example shown in Figure 3, the relations ANT1 and ANT2, as well as SYN1 and SYN2, are selected via the ***direct-motivated*** method (Method 1). The goal of this method is to locate a direct antonym *a1* of *t*, which has a synonym *s1* (or hypernym/hyponym *h1*) which is itself in an antonymy relation with a synonym *s2* (or hypernym/hyponym *h2*) of *a1*. The other relations in the network are then chosen via the ***direct-random*** method (Method 2), where the algorithm tries to find nodes in the GermaNet network that are directly attached to *t* and *a1* via the required relation types (in this case HYP1, COHYP1 and HYP2, COHYP2). If several nodes are available, a random choice is carried out. Thus, in Figure 4, the synonymy relations SYN1 and SYN2 as well as the antonymy relations ANT1 and ANT2 were established via the *direct-motivated* method in our algorithm, while HYP1, COHYP1 and HYP2, COHYP2 were established randomly via the *direct-random* method.

Methods 1 and 2 aim to maximise network density: By choosing synonyms of *t* and *a1* that are themselves related via antonymy, Method 1 aims to increase the density of the resulting net. On the other hand, Method 2 also works towards a close-knit net by choosing relations that are directly attached to *t* and *a1*. In addition, a special procedure applies to the direct-random choice of hypernyms and co-hyponyms: To increase the connectivity of the SemrelNet, preference is given to co-hyponyms and hypernyms of the target (cf. *c1* and *h1* in Figure 3) which are themselves related via a hypernymy relation (as is the case in Figure 4, where the dotted lines indicate a hypernymy relation). For this purpose, the algorithm first chooses a random co-hyponym of the target (but excluding lexical units which are simultaneously synonyms or antonyms of the target), and then includes the corresponding hypernym (if several are available, a random one is selected). Reversing the procedure by randomly choosing a hypernym first and then selecting one of its hyponyms as co-hyponym of the target would result in low probabilities for co-hyponyms with many siblings. Finally, while artificial concepts (cf. Section 3) are generally excluded from consideration as SemrelNet members, they are allowed as common hypernym of a target and its co-hypernym. Therefore, in cases where the corresponding hypernym turns out to be an artificial node in GermaNet, the co-hyponym is still selected, but another (non-artificial) hypernym or hyponym is randomly determined for the HYP relation. Figure 5 shows an example of a SemrelNet where COHYP2 involves an artificial common hypernym (*geschwindigkeitsspezifisch*, 'speed-specific'). HYP2 was determined in a second step via the *direct-random* method, which located a direct hyponym of *langsam* ('slow'): *schleppend* ('sluggish').

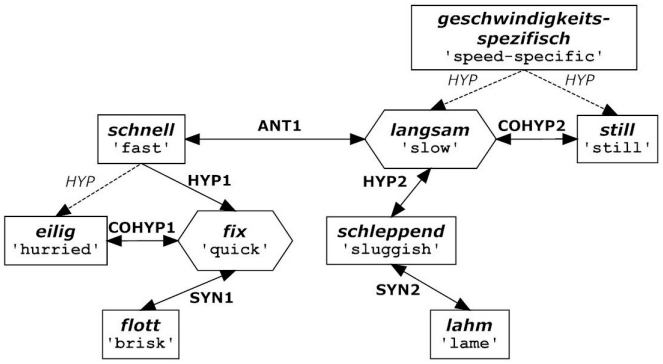


Figure 5: SemrelNet for target *fix* ('quick')

Figure 5 illustrates the situation where the first two methods fail because *t* does not have a direct antonym *a1*. This is where Method 3 (***indirect-random***) comes in: If no direct relations are available, the algorithm checks if any of the already existing nodes in the respective subnetwork (i.e. nodes which have already been filled by previous methods) are involved in one or more relations of the required type. If a match is found, a randomly-chosen relation and its associated node are added to the SemrelNet. The order in which existing nodes are checked is synonyms (1.), hypernyms/hyponyms (2.), and finally co-hyponyms (3.). For example, while SYN1, HYP1, and COHYP1 in Figure 5 were chosen via the direct-random mode, both ANT1 (attaching to the hypernym of the target of N_1, *schnell* 'fast') and SYN2 (attaching to the hyponym of the target of N_2, *schleppend* 'sluggish') were retrieved via the indirect-random method.

Finally, a back-off strategy was implemented to check for relations involving nodes that are directly connected to the target but not included as existing nodes in the given SemrelNet (Method 4, ***broken-random***). This means that there is no existing path in the network between the target and the (randomly selected) node, as illustrated in Figure 6. Here, SYN2 was chosen via the broken-random mode: The lexi-

cal unit *versauen* (‘to blow sth.’) has been marked as synonym of *vermasseln* (‘to mess up’), which is a hyponym of the target *durchfallen* (‘to fail (a test/exam)’). However, this hypernymy relation is not itself part of the network N, resulting in a broken path between *durchfallen* and *vermasseln*.

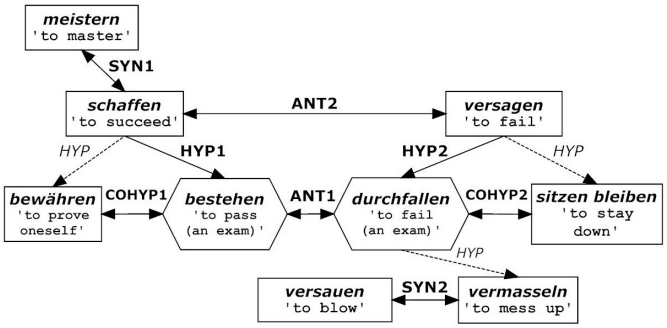


Figure 6: SemrelNet for target *bestehen* (‘to pass’)

Depending on the density of the network surrounding *t*, any number of nodes and associated relations in N may be blank: For example, if *t* and *a1* had no synonyms, the nodes *s1* and *s2*, as well as the relations SYN1/SYN2/ANT2, would be missing from the diagram in Figure 3. Similarly, if no antonym *a1* can be found for the members of N_1, subnet N_2 remains completely blank.

5.2. SemrelNet extraction tool

The algorithm described in the previous section has been implemented in Java and directly draws on the latest version of the GermaNet Java API (6.0.1)[6], which provides access to all information in GermaNet 6.0. A number of new classes and methods were implemented centering around the new concept ‘SemrelNet’. Instances of the SemrelNet class consist of a number of nodes (representing any participating lexical units in the SemrelNet, such as *s1*, *s2*, *h1*, *h2*, etc. as shown in Figure 3) and relations (linking a pair of nodes). For example, in the SemrelNet for target *unentschieden* (‘undecided’, cf. Figure 4), node *t* is realised by the lexical unit *unentschieden*, *s1* is realised by *unentschlossen* (‘undetermined’), and SYN1 links *t* and *s1*. In addition to their function in the net and the lexical unit which realises them, instances of the node class further record information about their position in the SemrelNet, relative to the target node *t*. For instance, node *s1* is typically involved in a synonymy relation within N_1, but due to the indirect-random and broken-random methods (cf. Section 5.1) it may appear in various positions within the subnet. For example, in Figure 6, *s1* (realised by *meistern*, ‘to master’) is an indirect synonym of *t*, being attached to the hypernym *h1* of *t*.
Table 3 provides an overview of the naming conventions used for node positions in a given SemrelNet, while Figure 7 shows the SemrelNet for target *bestehen* (‘to pass’) (cf. Figure 6) with added node labels of the format ‘function: position’. The labels show, for instance, that the node containing the lexical unit *sitzen bleiben* (‘to stay down (at school)’) has the function ‘c2’ (co-hypernym 2) and the position ‘cat’ (‘co-hypernym of antonym of t’). The position information on the *s2* (synonym 2) node with lexical unit *versauen* (‘to blow sth.’) indicates that there is a ‘broken’ path between *a1* and its hyponym *vermasseln* (‘to mess up’): In this case, ‘sUat’ reads as ‘synonym of broken hyponym of antonym of t’). Providing position information as shown in Table 3 is crucial for the graphical visualisation of SemrelNets.

Position	Read as...
t	target
sx / Sx	synonym / ‘broken’ synonym of x
ax / Ax	antonym / ‘broken’ antonym of x
ox / Ox	hypernym / ‘broken’ hypernym of x
ux / Ux	hyponym / ‘broken’ hyponym of x
cx / Cx	co-hyponym / ‘broken’ co-hyponym of x

Table 3: Naming conventions for SemrelNet node positions

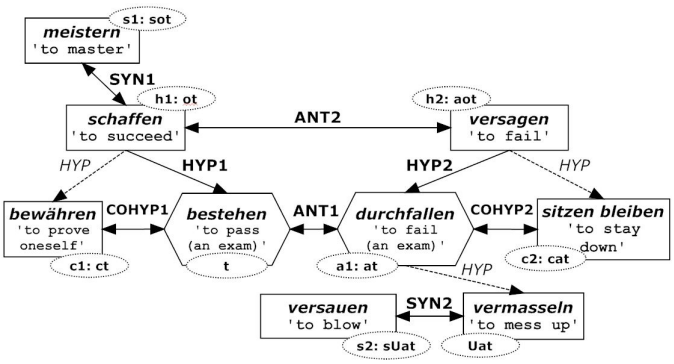


Figure 7: SemrelNet for target *bestehen* (‘to pass’) with added node position labels

The SemrelNet extraction tool produces two kinds of output: a simple text-based format (Figure 8) and XML format (Figure 9). In addition to listing the nodes and relations included in the nets, the output also provides information in terms of the GermaNet-IDs of all lexical units (attribute ‘id’ in Figure 9), and for each SemrelNet information about the target’s part of speech (attribute ‘pos’), semantic category (‘cat’), number of senses (‘senses’), corpus frequency (‘freq’), depth in the GermaNet hierarchy (‘depth’), and an overview of the completeness of the net (‘statsCode’ and ‘completeness’).
The SemrelNet extraction tool is freely available[7] and can be run on the whole of GermaNet, or on a selected list of lexical units. Due to the random methods included in the algorithm the resulting SemrelNets may be different when the tool is re-run several times on the same input data.

5.3. Results and discussion

This subsection intends to give an overview of the results of running the tool on the SemrelTargets dataset described

[6]http://www.sfs.uni-tuebingen.de/lsd/javadoc6.0/index.html

[7]http://www.ims.uni-stuttgart.de/projekte/semrel/resources.html

```
bestehen_76346 | V | Gesellschaft | 7 | 334228 | [4] | 1-1-1-1-1-1-1-1 | 8
ANT1: Dir-Motiv-AntR1 t:at | bestehen_76346 : durchfallen_76372
SYN1: Indir-Rand-Syn-T ot~sot | schaffen_76343 ~ meistern_76344
HYP1: Dir-Motiv-HypR1 t<ot | bestehen_76346 < schaffen_76343
COHYP1: Dir-Motiv-HypR1 t--ct | bestehen_76346 -- bewähren_76365
    ----> CH: schaffen_76343
----
ANT2: Dir-Motiv-HypR1 ot:oat | schaffen_76343 : versagen_76369
SYN2: Broken-Rand-Syn-A Uat~sUat | vermasseln_76375 ~ versauen_76376
HYP2: Dir-Motiv-HypR1 at<oat | durchfallen_76372 < versagen_76369
COHYP2: Dir-Motiv-HypR1 at--cat | durchfallen_76372 -- sitzen bleiben_76373
    ----> CH: versagen_76369
```

Figure 8: Text-based output of SemrelNet extraction tool

```
<relnet t="bestehen" id="76346" pos="V" cat="Gesellschaft" senses="7"
freq="334228" depth="4" statsCode="1-1-1-1-1-1-1-1" completeness="8">
<relation type="ANT1" rule="Dir-Motiv-AntR1">
<lu pos="t" id="76346">bestehen</lu>
<lu pos="at" id="76372">durchfallen</lu></relation>
<relation type="SYN1" rule="Indir-Rand-Syn-T">
<lu pos="ot" id="76343">schaffen</lu>
<lu pos="sot" id="76344">meistern</lu></relation>
<!-- remaining relations omitted for space reasons -->
</relnet>
```

Figure 9: XML output of SemrelNet extraction tool

in Section 4. Table 4 shows the size of the SemrelNets generated for the individual word classes. Complete nets (i.e. nets containing two instances of each of the four semantic relations ANT, SYN, HYP, and COHYP) are achieved for two thirds of all input adjectives (66), one third of verbs (32), but only for around one fifth of all input nouns (18). This is due to the fact that fewer nouns are involved in antonymy relations, which results in a large number of missing subnets N_2 (cf. Figure 3). As a consequence, the noun dataset contains a large number of SemrelNets of size 3 (59 altogether), typically containing the relations SYN1, HYP1, and COHYP1 (Figure 10).

Relations per net	Adj	N	V	All
1	0	0	0	0
2	4	4	4	12
3	21	59	47	127
4	0	0	0	0
5	1	0	0	1
6	1	0	1	2
7	6	18	15	39
8	66	18	32	116

Table 4: Number of relations per SemrelNet

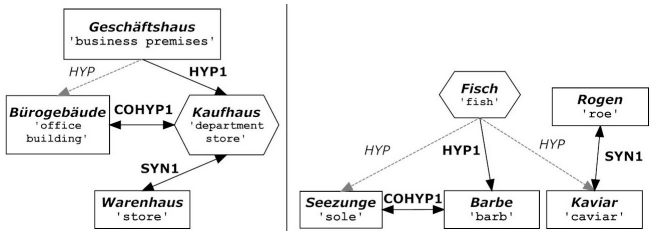


Figure 10: Examples of SemrelNets with three relations

With the exception of one example, which contains SYN1 and COHYP1 only, all 12 SemrelNets with only two relations include a HYP1 and COHYP1 relation (examples are shown in Figure 11). This is due to GermaNet's focus on the hypernymy hierarchy, which means that, generally, hypernyms and co-hyponyms are available for most lexical entries. All SemrelNets with three relations are of the type SYN1-HYP1-COHYP1 (as illustrated in Figure 10).

There are no SemrelNets with 4 relations, which again follows from GermaNet's structure as hypernym hierarchy: As soon as an antonym relation ANT1 is available, the paired lexical unit (referred to as *a1* in Figure 3) is likely to be involved in a hypernymy (HYP2) and/or co-hypernymy (COHYP2) relation. In other words, if a SemrelNet contains four relations, it will automatically contain a minimum of five relations altogether. Finally, it is worth noting that most instances of SemrelNets with seven relations (36 of 39) are missing an antonym relation, because antonymy is underrepresented across word classes (Figure 12).

Table 5 lists the total number of relation types included in the dataset. As expected, with the exception of one adjective, all input targets have SemrelNets which contain HYP1 and COHYP1 relations. The table further shows that an 'opposing' subnet N_2 exists for 74 adjectives (75%), 36 nouns (36.4%), and 48 verbs (48.5%, cf. row 'ANT1'). All

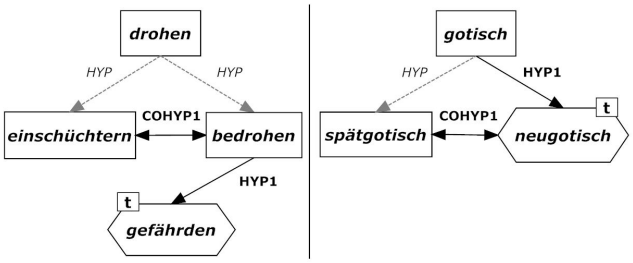


Figure 11: Examples of SemrelNets with two relations

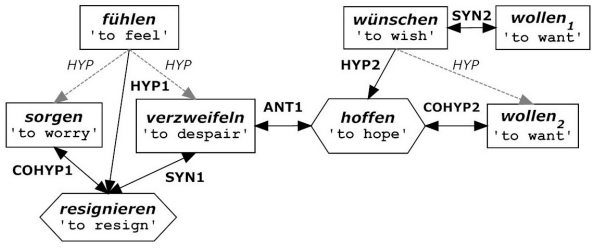


Figure 12: Example of SemrelNet with seven relations

N_2 include HYP2 and COHYP2 relations (with the exception of one adjective). Almost equally complete are the synonym relations: Only four adjectives, four nouns, and six verbs have no SYN1 relation in their network, and nearly all SemrelNets with a subnet N_2 also include a SYN2 relation (73 of 74 for adjectives, 36 of 36 nouns, and 47 of 48 verbs). The relation that fares worst in these statistics is ANT2: Only 18.2% (18) of noun targets, and 34.3% (34) of verb targets have a SemrelNet which includes ANT2. As noted above, this is due to the fact that (particularly for nouns) only a small number of antonym relations are encoded in GermaNet, and the chances of finding two of them within the same SemrelNet are therefore low. The situation is slightly better for adjective targets: Here, 67.7% (67) of SemrelNets contain an ANT2 relation. This is not surprising, since antonymy is considered the central organising principle for the adjectives in WordNets (Miller, 1990).

Relation	Adj	N	V	Total
ANT1	74	36	48	158
SYN1	95	95	93	283
HYP1	98	99	99	296
COHYP1	99	99	99	297
ANT2	67	18	34	119
SYN2	73	36	47	156
HYP2	73	36	48	157
COHYP2	73	36	48	157
TOTAL	652	455	516	1623

Table 5: Total number of relation types per word class

Finally, Table 6 gives an overview of how often the four extraction methods (described in Section 5.1) were employed in running the SemrelNet extraction tool on the input. The numbers show that the *direct-random* method is the most frequent by far, generating 66.3% of all relations (1076 of 1623). This supports the overall goal of making SemrelNets as random as possible, while still maintaining close density within the nets (by attaching relations directly to the target nodes). In contrast, the *direct-motivated* rules, whose aim is to maximise connectivity by detecting a second antonymy link between subnets N_1 and N_2, were only triggered 110 times for all word classes, being most frequently used for adjectives (71 times). The second most frequent method in all word classes is the *indirect-random* one with 15.5% for adjectives (101/652), 14.1% for nouns (64/455), and 16.1% for verbs (83/516). The use of this method results in a lower density of the net, as the selected relations are only indirectly attached to the target. However, the method still supports connectivity of the nets, as the relations are attached to other existing nodes in the net. The back-off strategy, in which so-called *broken-random* relations are considered, is used least frequently among the random relations for all word classes, with 10.0% of all adjective relations (65), 11.0% of all noun (50), and 11.4% of all verb relations (59) having been triggered by this method. Of the resulting broken-random relations included in the dataset, more than half are antonyms (56.9%, 99/174), 36.8% synonyms (64/174), and 6.3% hypernyms (11/174).

Method	Adj	N	V	Total
Direct-motivated	71	8	31	110
Direct-random	409	332	335	1076
Indirect-random	101	64	83	248
Broken-random	65	50	59	174
Other	6	1	8	15
TOTAL	652	455	516	1623

Table 6: Number of methods employed per word class

6. Current and future work

The datasets described in the previous sections are currently being used in two types of human judgement experiments: One focusing on the generation of semantically related word pairs, and the other on human rating of the strength of semantic relations. Both experiments are hosted on Amazon Mechanical Turk (MTurk)[8].

The purpose of the first experiment is to gather human associations for each type of semantic relation. That is, for each lexical unit in the SemrelTargets dataset, participants are asked to generate one synonym, one antonym, one hypernym, and one co-hyponym. In order to avoid confusion between the different types of relation, the data is presented to participants in bundles of 11 words (or 11 "HITs", as individual decision tasks are called in MTurk) to be assessed for the same type of relation (e.g. finding antonyms for each of the 11 words). The goal is to receive associations from at least 10 different participants for each target. To make sure that the data is dealt with properly, and to exclude non-native speakers of German, each set of 11 HITs includes two examples of non-words, which should be recognised

[8] www.mturk.com

as such by native speakers of German (e.g. *Blapselheit*, *gekortiert*). If not, the whole set is excluded.
In the second experiment, participants are presented with word pairs included in the SemrelNets dataset, and asked to rate their degree of synonymy, antonymy, etc. on a scale between 0 and 5, plus an option for marking unknown words. Again, to avoid confusion between the different types of relation, each bundle of 14 HITs is rated according to one specific relation at a time. Each bundle contains:

1. 3 focus-relation pairs (i.e. the relation under consideration)
2. 9 other-relation pairs (i.e. 3 pairs each from the other three relations)
3. 2 test pairs (involving one nonsense-word)

Once the experiments are completed, each word pair in the SemrelNets database will have received 10 ratings each for their degree of synonymy, antonymy, hypernymy, and co-hyponymy.

7. Conclusion

This paper described the design and compilation of a new dataset containing semantically coherent relation pairs drawn from GermaNet. The dataset consists of two parts:

1. Three sets of 99 lexical units drawn from the three major word classes adjectives, nouns, and verbs, using a stratified sampling technique to balance the datatset for semantic category, polysemy, and type frequency ('SemrelTargets'); and
2. Three sets of 99 semantically coherent subnets clustering around the SemrelTargets, and consisting of a total of 1623 paradigmatic semantic relation pairs ('SemrelNets').

The data is currently being used in two human judgement experiments, in which (1) new relation pairs are generated from the set of SemrelTargets, and (2) word pairs in the SemrelNets are rated for the strength of the semantic relations holding between them. The dataset thus promises to be an important resource not only for research in computational linguistics, but also for studies in theoretical linguistics and psycholinguistics.

8. References

Sharon A. Caraballo. 2001. *Automatic Acquisition of a Hypernym-labeled Noun Hierarchy from Text*. Ph.D. thesis, Brown University.

Alan Cruse. 1986. *Lexical Semantics*. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge, UK.

James Curran. 2003. *From Distributional to Semantic Similarity*. Ph.D. thesis, Institute for Communicating and Collaborative Systems, School of Informatics, University of Edinburgh.

Marie-Catherine de Marneffe, Anna N. Rafferty, and Christopher D. Manning. 2008. Finding Contradictions in Text. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1039–1047, Columbus, OH.

Philip Edmonds and Graeme Hirst. 2002. Near-Synonymy and Lexical Choice. *Computational Linguistics*, 28(2):105–144.

Christiane Fellbaum, editor. 1998. *WordNet – An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Iryna Gurevych. 2005. Using the Structure of a Conceptual Network in Computing Semantic Relatedness. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing*, pages 767–778, Jeju Island, Korea.

Iryna Gurevych. 2006. Thinking beyond the Nouns - Computing Semantic Relatedness across Parts of Speech. In *Sprachdokumentation & Sprachbeschreibung, 28. Jahrestagung der Deutschen Gesellschaft für Sprachwissenschaft*, Bielefeld, Germany.

Sanda M. Harabagiu, Andrew Hickl, and Finley Lacatusu. 2006. Negation, Contrast and Contradiction in Text Processing. In *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 755–762, Boston, MA.

Marti Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *Proceedings of the 14th International Conference on Computational Linguistics*, Nantes, France.

Jay J. Jiang and David W. Conrath. 1997. Semantic Similarity based on Corpus Statistics and Lexical Taxonomy. In *Proceedings on International Conference on Research in Computational Linguistics*, pages 19–33.

Claudia Kunze and Andreas Wagner. 1999. Integrating GermaNet into EuroWordNet, a Multilingual Lexical-Semantic Database. *Sprache und Datenverarbeitung*, 23(2):5–19.

Lothar Lemnitzer and Claudia Kunze. 2007. *Computerlexikographie*. Gunter Narr Verlag, Tübingen, Germany.

Dekang Lin, Shaojun Zhao, Lijuan Qin, and Ming Zhou. 2001. Identifying Synonyms among Distributionally Similar Words. In *Proceedings of the International Conferences on Artificial Intelligence*, pages 1492–1493, Acapulco, Mexico.

Cupertino Lucerto, David Pinto, and Héctor Jiménez-Salazar. 2004. An Automatic Method to Identify Antonymy Relations. In *Proceedings of the IBERAMIA Workshop on Lexical Resources and the Web for Word Sense Disambiguation*, pages 105–111, Puebla, Mexico.

John Lyons. 1977. *Semantics, Volume II*. Cambridge University Press, Cambridge, UK.

George A. Miller and Walter G. Charles. 1991. Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes*, 6(1):1–28.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to Wordnet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4):235–244.

George A. Miller, editor. 1990. *WordNet: An On-line Lex-*

ical Database, volume 3 (4). Oxford University Press. Special Issue of the International Journal of Lexicography.

Jane Morris and Graeme Hirst. 2004. Non-Classical Lexical Semantic Relations. In *Proceedings of the HLT Workshop on Computational Lexical Semantics*, Boston, MA.

M. Lynne Murphy. 2003. *Semantic Relations and the Lexicon*. Cambridge University Press.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 113–120, Sydney, Australia.

Philip Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1*, pages 448–453, San Francisco, CA.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual Correlates of Synonymy. *Communications of the ACM*, 8:627–633.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2004. Learning Syntactic Patterns for Automatic Hypernym Discovery. *Advances in Neural Information Processing Systems*, 17.

Peter D. Turney. 2006. Expressing Implicit Semantic Relations without Supervision. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 313–320, Sydney, Australia.

Torsten Zesch and Iryna Gurevych. 2006. Automatically Creating Datasets for Measures of Semantic Relatedness. In *COLING/ACL 2006 Workshop on Linguistic Distances*, pages 16–24, Sydney, Australia.

The SIMILAR Corpus: A Resource To Foster The Qualitative Understanding of Semantic Similarity of Texts

Vasile Rus, Mihai Lintean, Cristian Moldovan, William Baggett, Nobal Niraula, and Brent Morgan

Department of Computer Science, Department of Psychology, Institute for Intelligent Systems
The University of Memphis
Memphis, TN 38152
E-mail: vrus@memphis.edu, mclinten@memphis.edu, cmldovan@memphis.edu, nbnraula@memphis.edu, writebill@gmail.com, brent.morgan@gmail.com

Abstract

We describe in this paper the SIMILAR corpus which was developed to foster a deeper and qualitative understanding of word-to-word semantic similarity metrics and their role on the more general problem of text-to-text semantic similarity. The SIMILAR corpus fills a gap in existing resources that are meant to support the development of text-to-text similarity methods based on word-level similarities. The existing resources, such as data sets annotated with paraphrase information between two sentences, do not provide word-to-word semantic similarity annotations and quality judgments at word-level. We annotated 700 pairs of sentences from the Microsoft Research Paraphrase corpus with word-to-word semantic similarity information using both a greedy and optimal protocol. We proposed a set of qualitative word-to-word semantic similarity relations which were then used to annotate the corpus. We also present a detailed analysis of various quantitative word-to-word semantic similarity metrics and how they relate to our qualitative relations. A software tool has been developed to facilitate the annotation of texts using the proposed protocol.

Keywords: word-to-word semantic similarity, paraphrase identification, entailment recognition

1. Paper

We describe in this paper our effort to fill a gap in existing resources for the study of semantic similarity of texts. We have designed a protocol and created an annotated data set to foster a deeper and qualitative understanding of word-to-word semantic similarity measures together with their role on the more general task of assessing the semantic similarity of texts (containing more than one word). An example of a text-to-text semantic similarity task is the task of paraphrase identification (Dolan, Quirk, and Brockett, 2004).

The semantic similarity approach, as a practical alternative to the full understanding approach to the task of natural language understanding (Rus & Lintean, submitted), has been successfully applied to a series of fundamental text-to-text similarity tasks in natural language processing: paraphrase identification (Dolan, Quirk, and Brockett, 2004), recognizing textual entailment (Dagan, Glickman, & Magnini, 2005; Rus & Graesser, 2006), and elaboration detection (McCarthy & McNamara, 2008). These fundamental tasks are in turn important to a myriad of real world applications such as providing evidence for the correctness of answers in Question Answering (Ibrahim, Katz, & Lin, 2003), increase diversity of generated text in Natural Language Generation (Iordanskaja, R. Kittredge, & A. Polgere, 1991), assessing the correctness of student responses in Intelligent Tutoring Systems (Graesser, Hu, McNamara, 2005), and identifying duplicate bug reports in Software Testing (Rus et al., 2009). Table 1 provides examples of text pairs from semantic similarity tasks proposed by various research groups over the last decade.

Much research has been dedicated to proposing word-to-word similarity metrics (Pedersen, Patwardhan, and Michelizzi, 2004) and more recently to developing methods to compute the semantic similarity of larger texts. Among the latter, a particular set of methods that address the larger text-to-text similarity problem are those that rely on word-level similarity metrics (e.g. the similarity of two sentences or paragraphs; Corley & Mihalcea, 2005; Lintean et al., 2010) and which we call compositional methods as they are based on the principle of compositionality. The compositional principle states that the meaning of longer texts can be composed from the meaning of its parts, i.e. words.

To the best of our knowledge existing methods to solve the text-to-text similarity problem using word-level similarities limit themselves to a quantitative analysis of the overall method's performance on a given text-to-text similarity task, e.g. paraphrase identification, as opposed to a more detailed quantitative and qualitative understanding of the word-to-word similarity metrics and their impact on the text-to-text similarity method proposed. How does the average similarity score between words that are deemed similar beyond any doubt compare to the average similarity score between words that are deemed similar in some context? For instance, what is the qualitative difference between a similarity score of 0.90 and a score of 0.70 (we assume normalized similarity scores only)? What about between a score of 0.45 and a score of 0.55? Also, it is not known at what extent these word-level metrics capture more than lexical information, e.g. context and world knowledge. We take a first step towards a better understanding of word-to-word similarity metrics and their actual impact on methods using these metrics.

To this end, we propose a protocol that maps existing

ID	Text 1 **(assumed to be True for tutoring and RTE data)**	Text 2	Source/Relation
1	**Expert Answer:** The force of the earth's gravity, being vertically down, has no effect on the object's horizontal velocity	***Student Input:*** *The horizontal component of motion is not affected by vertical forces*	AutoTutor/True Paraphrase
2	**Textbook Sentence**: A glacier's own weight plays a critical role in the movement of the glacier.	**Student Input**: A glacier's movement depends on its weight.	iSTART/True Paraphrase
3	The procedure is generally performed in the second or third trimester.	*The technique is used during the second and, occasionally, third trimester of pregnancy.*	MSR/True Paraphrase
4	**Text**: Deployment of Filipino workers in Iraq suspended by Philippine president due to repeated kidnappings.	**Hypothesis**: *Filippino workers have been kidnapped by the Philippine president.*	RTE/False Entailment

Table 1. *Examples of text pairs from four different datasets: AutoTutor, iSTART, Microfost Research Paraphrase (MSR) corpus, and Recognizing Textual Entailment (RTE) corpus.*

word-to-word similarity metrics onto qualitative judgments of similarity such as CLOSE (the words are similar beyond any doubt, e.g. *student* and *learner*), RELATED (the words are related but they are not quite similar, e.g. *boxing* and *fight*), CONTEXT (the words are matched within the context of the texts to be assessed, e.g. *totalling* and *volume* – see the whole context later), and KNOWLEDGE (world or domain knowledge is needed to match the words, e.g. *retailer* and *WalMart*). These qualitative judgments are then related to existing quantitative word-to-word similarity metrics for a better understanding and interpretation of the metrics.

The protocol was designed in the context of qualitative assessments of the similarity of two texts. That is, judges were shown two texts which might or might not be semantically similar, e.g. paraphrases, and asked to match words and indicate the reason such as CLOSE, RELATED, CONTEXT, KNOWLEDGE. A default NONE value is assigned to unmatched words. Identical words (in their raw form) in both sentences were deemed perfectly similar and annotated automatically with the label IDENTICAL.

We chose as our starting data set the Microsoft Research Paraphrase corpus (MSRP; Dolan, Quirk, and Brockett, 2004) used to evaluate methods addressing the task of paraphrase identification. The corpus has been widely used by many research groups (Corley & Mihalcea, 2005; Lintean & Rus, 2009; Lintean et al., 2010) and therefore would allow us to compare the results of word matching by human annotators with the machings proposed by the automated methods. We have asked the human experts to pair words greedily as well as optimally. The greedy annotation was necessary in order to emulate existing automated greedy methods (Corley and Mihalcea, 2005; Lintean et al., 2010) which would allow for a direct comparison with human greedy judgments. In the greedy annotation, we asked humans to consider one word at a time in one text, say T1, and greedily match it to a word in the other text, T2, without considering the whole text T1 as a context. Optimal annotation of similar words was based on human judges' full understanding of the texts.

We annotated as of this writing 700 pairs of sentences from the MSRP corpus which consists of 29,771 tokens (words and punctuation) of which 26,120 are true words and 17,601 content words. The 700-pair dataset also contains 12,560 true relations (a true relation is of any type except NONE) identified when greedily identifying similarities from T1 to T2 (target words were selected from T1) and 12,345 true relations identified when greedily annotating from T2 to T1. For the optimum annotation, 15,692 relations were identified. We report a detailed analysis of the so obtained corpus, called the SIMILAR corpus, and compare the human annotations with results obtained by matching words using the word-to-word semantic similarity measures in the WordNet Similarity library (Pedersen, Patwardhan, and Michelizzi, 2004) as well as using Latent Semantic Analysis (LSA; Landauer et al., 2007).

A semantic annotation tool was also developed that allowed our experts to easily annotate the SIMILAR corpus. The tool offers an user-friendly interface which tremendously speeds up the transfer of the proposed annotation protocol to new texts, in any language, and also offers great productivity advantage allowing for annotating more text per unit of time. If the paper is accepted, both the corpus and the annotated data set will be available at our website: HIDDEN.

The rest of the paper is organized as in the following. The next section presents related work on semantic similarity with an emphasis on compositional approaches based on word-to-word similarity metrics. Section 3 describes in details the guidelines for greedy annotation while section 4 presents guidelines for optimum annotation. The annotation tool is briefly described in section 5. The details of the SIMILAR corpus are presented in the following section. The Conclusions section ends the paper.

2. Related Work

Assessing the semantic similarity of texts has been explored at different levels of granularity: word-to-word, sentence-to-sentence, paragraph-to-paragraph (Rus, Lintean, & Azevedo, 2009), and document-to-document (see Information Retrieval work; Salton, Wong, & Yang, 1975). We focus next on word-to-word similarity and sentence-to-sentence similarity work as it is most relevant to ours.

Word-to-word similarity research culminated with the release of the WordNet similarity package by Pedersen, Patwardhan, and Michelizzi (2004). Other notable work that allows quantifying how similar words are is the Latent Semantic Analysis framework (described below) and more recently Latent Dirichlet Allocation (LDA; Blei, Ng, & Jordan, 2003). Other frameworks exists which we do not mention due to space limitations.

Extending word-to-word similarity measures to sentence level and beyond has drawn increasing interest in the last decade or so in the Natural Language Processing community. The interest has been driven primarily by the creation of standardized data sets and corresponding shared task evaluation campaigns (STECs) for the major text-to-text qualitative semantic relations of entailment (RTE; Recognizing Textual Entailment corpus by Dagan, Glickman, & Magnini, (2005), paraphrase (MSRP; Microsoft Research Paraphrase corpus by Dolan, Quirk, and Brockett, 2004), and elaboration (ULPC; User Language Paraphrase Challenge by McCarthy & McNamara, 2008).

Assessing the semantic similarity of two texts, T1 and T2, using a compositional approach based on word-to-word semantic similarity metrics has been primarily approached using greedy methods (Corley & Mihalcea, 2005; Lintean & Rus, 2009; Lintean et al., 2010) and more recently an optimal method (Rus & Lintean, in press). We briefly describe these approaches as they are relevant to our corpus annotation effort.

Corley and Mihalcea (2005) presented one of the earliest methods to compute the similarity of two sentences using word-to-word similarity methods. In their method, they computed the similarity of two texts by greedily summing up the maximum similarity of each word in one sentence to any word in the opposite sentence. The individual word-to-word similarities were computed using measures from the WordNet similarity package (Pedersen, Patwardhan, & Michelizzi, 2004) as well as a simple vector space model. They report results on the MSRP corpus. Other notable work is by Rus and colleagues (2008) who addressed the task of paraphrase identification using the MSRP corpus by computing the degree of subsumption at lexical and syntactic level between two sentences in a greedy manner as well.

Assessing the correctness of student contributions in dialogue-based tutoring systems has been approached either as a paraphrase identification task (Graesser, Hu, McNamara, 2005; Graesser, Olney, et al., 2005), i.e. the task was to assess how similar student contributions were to expert-generated answers, or as an entailment task (Rus & Graesser, 2006), in which case the task was to assess whether student contributions were entailed by expert-generated answers. The expert answers were assumed to be true. If a correct expert answer entailed a student contribution then the contribution was deemed to be true as well.

Latent Semantic Analysis (LSA; Landauer et al., 2007) has been used to evaluate student contributions during the dialog between the student and a dialogue-based tutoring system (Graesser, Hu, & McNamara, 2005; VanLehn et al., 2007). In LSA the meaning of a word is represented by a reduced-dimensionality vector derived by applying an algebraic method, called Singular Value Decomposition (SVD), to a term-by-document matrix built from a large collection of documents. A typical dimensionality of an LSA vector is 300-500 dimensions. To compute the similarity of two words the cosine of the words' corresponding LSA vectors is computed (cosine is the normalized dot-product). A typical extension of LSA-based word similarity to computing the similarity of two sentences (or even larger texts) is to use vector algebra to generate a single vector for each of the sentences/texts (by adding up the LSA vectors of the individual words) and then compute the cosine between the resulting sentence/text vectors. Another approach proposed, greedily selects for each word its best match using the cosine of the words' LSA vectors, and then sums the individual word-to-word similarities in order to compute the overall similarity score for the two sentences (Lintean et al., 2010). Our work is mostly relevant to LSA-based approaches using only the latter method as it is the only approach that fits with a compositional model based on word-to-word similarity.

We describe the greedy and optimal methods in more details next. It is important to describe them as our manual annotation tries to emulate them (although the optimal manual annotation is slightly different compared to the optimal automated method).

Greedy Method

In the greedy method, each word in text T1 is paired with every word in text T2 and word-to-word similarity scores are computed according to some metric. For each word in T1, its best matching word in T2 is greedily retained. These greedily-obtained scores are added up using a simple or weighted sum which can be normalized in different ways, e.g. by dividing to the longest text or to the average length of the two texts. The formula we show here is given in equation 1 (from Lintean & Rus, 2009). As one would notice, this formula is asymmetric, i.e. score(T1,T2) ≠ score(T2,T1). The average of the two scores provides a symmetric similarity score, more suitable for a paraphrase task, as shown in Equation 2. Given that identical words occurring in the two texts are perfectly matched, the greedy method matches identical words first.

$$score(T1,T2) = \frac{\sum_{v \in T1} weight(v) * \max_{w \in T2} word - sim(v,w)}{\sum_{v \in T1} weight(v)}$$

Equation 1. *Asymmetric semantic similarity score between texts T1 and T2.*

$$simScore(T1,T2) = \frac{score(T1,T2) + score(T2,T1)}{2}$$

Equation 2. *Symmetric semantic similarity score between*

ID	SENTENCE	TARGET	SEMANTIC RELATION
1	In Nigeria alone, the report estimated that between 100,000 and 1 million girls and women are suffering from the condition.	running	NONE
2	The charges allege that he was part of the conspiracy to kill and kidnap **persons** in a foreign country.	individual	WORD
3	Hearing was partially **restored** by an electronic ear implant.	regained	WORD
4	In Nigeria alone, the report said, as many as 1 million women may be **living with the condition**.	suffering	PHRASE
5	Jeter, who **dislocated his left shoulder** in a collision March 31, took batting practice on the field for the first time Monday.	injury	PHRASE
6	NASA satellite images show that Arctic ice has been shrinking at the rate of nearly 10 percent a decade.	disappearing	CONTEXT
7	Duke and North Carolina have been resolute in their positions against expansion.	oppose	CONTEXT
8	The retailer said it came to the decision after hearing the opinions of customers and associates.	Wal-Mart	WORLD KNOWLEDGE
9	Duke and North Carolina have been resolute in their positions against expansion.	school	WORLD KNOWLEDGE

texts T1 and T2.

Table 2. *Examples of target words (third column), opposite sentences (column two), and qualitative similarity relations (last column).*

The obvious drawback of the greedy method is that it does not aim for a global maximum similarity score. The optimal method (Rus & Lintean, in press) which is described next solves this issue.

Optimal Method

The optimal matching solution (Rus & Lintean, in press) was inspired by the optimal assignment problem which is one of the fundamental combinatorial optimization problems and consists of finding a maximum weight matching in a weighted bipartite graph.

Given a weighted complete bipartite graph $G = X \cup Y; X \times Y)$, where edge xy has weight $w(xy)$, find a matching M from X to Y with maximum weight.

A famous instance of the optimal assignment problem is job assignment which is about assigning a group of workers, e.g. sailors, to a set of jobs (on ships) based on the expertise level, measured by $w(xy)$, of each worker at each job (Dasgupta et al., 2009). By adding dummy workers or jobs we may assume that X and Y have the same size, n, and can be viewed as $X = \{x_1, x_2, \dots, x_n\}$ and Y $=\{y_1, y_2, \dots, y_n\}$. In the semantic similarity case, the workers and jobs are words from the two sentences to be compared and the weight $w(xy)$ is the word-to-word similarity between words x and y in the two sentences, respectively.

The assignment problem can thus be formulated as finding a permutation π of {1, 2, 3, … , n} for which $\sum_{i=1}^{n} w(x_i y_{\pi(i)})$ is maximum (Dawes, 2011). Such an assignment is called optimum assignment. An algorithm, the Kuhn-Munkres method (Kuhn, 1955), has been proposed that can find a solution to the optimum assignment problem in polynomial time. It is beyond the scope of this paper to present the details of the algorithm.

The method guarantees optimal overall best match. That is, Rus and Lintean (in press) showed how using the Kuhn-Munkres algorithm words in text T1 (the sailors) cab be optimally matched to words in text T2 (the ships) based on how well the words in T1 (the sailors) fit the words in T2 (the ships). The fitness between the words is nothing else but their word-to-word similarity according to some metric of word similarity.

Based on these two categories of compositional semantic similarity approaches that rely on word-to-word similarity metrics, greedy and optimal, we have designed two annotation protocols: greedy and optimal annotation.

3. Greedy Word-to-Word Annotation

As already mentioned, the greedy matching strategy was inspired from automated greedy methods proposed for the task of semantic similarity of short texts. The greedy methods pair a target word in one sentence with all the words in the other sentence and retain the matching word with the highest word-to-word similarity score to the target word regardless of how other words match each other.

If human judges were to emulate this process they would have to consider one individual word from one sentence, called the target word, and try to find a best matching word in the other sentence regardless of how other words would match. This isolation assumption is needed to emulate the word-to-word similarity measures as closely as possible and allow a direct comparison between human judgments and automated methods. Table 2 illustrates how the greedy annotation occurred. It also provides examples for each type of qualitative word-to-word relations we defined. The third columns shows target words, from text T1, and the second column all candidates words from text T2. The other words in T1 are irrelevant in greedy matching. Note the greedy matching needs to be performed in two phases. Phase one means selecting target words from T1 and find best matches in T2. Phase two involves selecting target words from T2 and find best matches in T1.

3.1 The Qualitative Word-to-Word Relations

When selecting the best matching individual word in the

opposite sentence for a given target word, judges must decide whether a matching word exist (or not). If a matching word exists, a judgment on the type of matching needs to be made. A matching word could be a word which is semantically close, based on judge's view, to the target word. Semantically close words are words that are synonyms such as *person* and *individual*, or deemed semantically close beyond any reasonable doubt by a human judge. If words have multiple senses, at least two senses of the two words are semantically close beyond any reasonable doubt). For instance, the words *research* and *study* are semantically close when considering their meaning of *investigating* a particular issue.

In case a semantically close word is not found, a word that is somehow semantically related should be chosen, e.g. *boxing* and *fighting* are semantically related but not semantically close.

These two types of annotations would be sufficient to directly evaluate greedy automated methods against the human greedy judgments. However, we wanted to go beyond that. We decided to include in the annotation protocol several additional types of qualitative semantic relations.

If a target word is not similarly close or related as defined above to any individual word in the other sentence (when considering these words in isolation), it might be the case that the two words could be deemed similar if the context of the matching word (but not of the target word) could help in relating semantically the words. For instance, the target word *totalling* is contextually related to *volume* in the second sentence below if considering the full *context* of the second sentence.

T1: Singapore is already the United States' 12th-largest trading partner, with two-way trade totaling more than $34 billion.

T2: Although a small city-state, Singapore is the 12th-largest trading partner of the United States, with trade volume of $33.4 billion last year.

For the context relation it might be the case that a particular target word cannot be matched against one individual word in the other sentence. It is rather the case that the other sentence entirely implies or suggests the target word in which case the target word is related to the context of entire sentence instead of one particular word. This might be the case also for the next type of relation, KNOWLEDGE.

Sometimes even context is not enough to relate a target word to any other word in the opposite sentence. Word knowledge could help. In the above example, when matching the target word/collocation *city-state* world knowledge is needed to relate it to *Singapore* in the first sentence.

Sometimes a target word, e.g. the collocation *credit_card* in the second sentence below, cannot be matched in any way to a word in the other sentence. In this case, the NONE relation is chosen for the target word.

T1: He said it was a mistake, and he reimbursed the party nearly $2,000.

T2: The governor said the use of the credit card was a mistake, and has since reimbursed the party for the expense.

3.2 Additional Guidelines

Collocations such as *give_up* or *joint_venture* were considered individual words because word-to-word similarity metrics consider them so and therefore similarity scores can be computed between collocations or between a collocation and a simple word.

Numbers were deemed as either semantically close, when identical, or semantically related when representing different values, e.g. 123 and 345 are related.

Temporal markers, such as *today* or *yesterday*, were deemed close, when identical, and related when different.

Pronouns should were deemed close, when identical, and contextually related to a referent when could be linked to the referent in the opposite sentence (or NONE if no reasonable referent was found).

Punctuation had to be matched to an identical punctuation mark in the opposite sentence.

Verbs were matched using their base forms and ignoring inflections. For instance, *go*, *went*, *gone* were all matched with each other.

Auxiliaries, e.g. *has* in *has gone*, were labelled with NONE if the main verb (i.e. *gone*) had no match in the opposite sentence. When the main verb does have a match, the auxiliary was matched with a matching/corresponding auxiliary in the opposite sentence.

Function words, e.g. *of* or *which*, that are in one sentence but not the other were labelled CONTEXT or NONE depending on the human rater's judgment with respect to how strong the function word is implied by the other sentence. Function words play more of a syntactic role, i.e. they are more relevant in a context. If a function word is present in one sentence and not the other than it can only be linked to the opposite sentence via CONTEXT at best (or NONE).

All tokens (words/collocations and punctuation) must be explicitly matched (even if choosing the NONE matching).

Importantly, in greedy matching many-to-one relations are possible. In the example below, when matching *Duke* to a token in the other sentence it will be matched with *school*. Similarly, when *school* in the first sentence is matched it will be matched with *school* in the second sentence. Therefore, *Duke* and *school* in the first sentence will be matched to the same word, *school*, in the second sentence.

T1: Duke spokesman expressed concerns about the school's financial security.

T2: School representative expressed concerns about the university's financial security.

4. Optimal Annotation

The optimal matching strategy is inspired from optimal matching methods proposed for tasks where a set of items must be matched against another set while optimizing the overall matching score and not individual scores. The

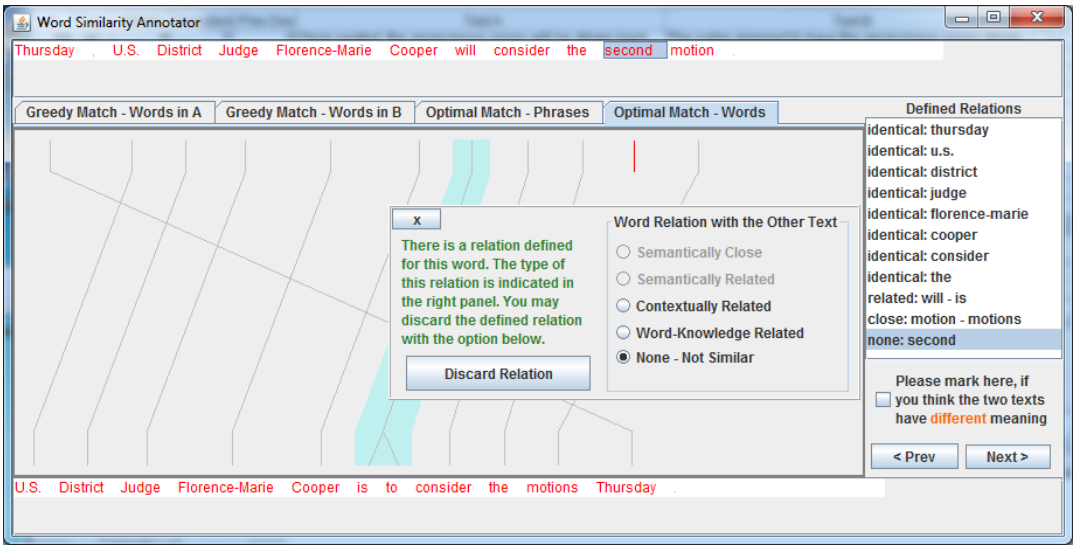


Figure 1. A snapshot of SIMILAT (SIMILar Annotation Tool).

overall matching score is the sum of individual scores for pairs of items, one from one set and the other item from the other set.

While in greedy matching the goal is for a target word to find a best matching word in the opposite sentence, in optimal matching the goal is to match items such that an overall optimal matching is achieved. Because it will be extremely time-consuming and error-prone to ask humans to fully emulate the optimal assignment algorithm, we simply asked them to pair words based on their full understanding of the two sentences. That is, given their reading of the two sentences judges were supposed to match the words that would make sense.

As opposed to greedy matching where one-to-many relations among words was possible, in optimal matching we strive for one-to-one matching.

An example of a pair of sentences where the greedy matching approach does not provide best overall, global match is given below.

T1: Duke spokesman expressed concerns about the school's financial security.

T2: School representative expressed concerns about the university's financial security.

In one matching, a target word, say *Duke* in the above example, can be greedily matched to the closest word in the other sentence, which is *university* (not *school*). In another matching, the target word *Duke* can be matched with the best matching word in the other sentence considering a more global assessment of both sentences. In our case, global matching would relate *Duke* with *school* and *school* in the first sentence with *university* in the second sentence.

Optimal matching involves matching words or phrases as best implied by the context of both sentences. Instead of focusing on a word, the focus is on finding the best match possible, which could be between two words, a word and a phrase, or two phrases. Optimal matching consists of two steps, as outlined below.

Step 1. Match chunks of the two sentences which are semantically equivalent beyond any doubts and whose equivalent meaning cannot be inferred from their words; that is, the meaning of these chunks could only grasped from the chunks as a whole; Examples of such semantically equivalent chunks/phrases are *give birth* and *have a child* or *have an offspring, living with the condition* and *suffering from a condition*.

Step 2. Eventually using information from Step 1, match individual words such that optimal matching is being achieved (at word-level). That is, a word should be matched against its best matching word as implied by the context of the two sentences and not necessarily its best individual match. For instance, a word should not be matched with an identical word in the opposite sentence if the context suggests the word should be matched to something else.

Examples of optimal matching are given below. The phrase *suffering from a condition* should be matched with the phrase *living with the condition* in the example below instead of just matching *suffering* with the word *condition* (based on individual similarities) or *suffering* with *living* (based on individual similarities and context).

T1: In Nigeria alone, the report said, as many as 1 million women may be living with the condition.

T2: In Nigeria alone, the report estimated that between 100,000 and 1 million girls and women are suffering from the condition.

For the pair of sentences below, the phrase *gives birth* and *has her first child* have the same meaning and therefore an optimal matching approach constrains the matching process to words within those phrases. That is, *birth* should only be matched to a word from the matching phrase *has her first child*.

T1: Crossing Jordan will be back in January after star Jill Hennessy gives birth.

T2: NBS also plans to shelve Crossing Jordan until January as star Jill Hennessy has her first child.

In this example below, no chunks should be selected as being equivalent because all chunks/phrases could be deemed similar (or not) based on their component words.

T1: The procedure is generally performed in the second or third trimester.
T2: The technique is used during the second and, occasionally, third trimester of pregnancy.

5. SIMILAT: The Semantic Annotation Tool

We have developed a tool to help our annotators easily annotate word-to-word relations. The annotation tool is called SIMILAT (SIMILarity Annotation Tool). A snapshot of the tool is shown in Figure 1.

The pair of two texts whose words are to be matched are shown at the top and bottom of SIMILAT's window. Below the text at the top, there are four tabs that support four different types of annotations: Greedy Match – Words in A, Greedy Match – Words in B, Optimal Match – Phrases, and Optimal Match – Words. Optimal Match – Phrases is a type of annotation that is currently under development and is not being described here. Greedy Match – Words in A allows the user to match one word at a time in the top text (called text A) to any word in the bottom text, called text B. This corresponds to the greedy annotation when target words are selected from text A. Similar, Greedy Match – Words in B allows the annotator to match one target word at a time in the bottom text to any word in text A. Optimal Match – Words facilitates optimal matching of words in which case any word in either text A or text B can be matched with a word and only one (or the whole context of the opposite sentence) or nothing in the other text. All the matchings can be done using the mouse by selecting the words to be matched and then choosing the type of relations from the pop-up menu: CLOSE, RELATED, CONTEXT, WORLD-KNOWLEDGE, and NONE. IDENTICAL matchings are automatically detected and shown in red.

As an annotator pairs certain words, they change their color to red to visually indicate they have been paired. The annotator must explicitly select a NONE relation for unmatched words so that they turn red. This assures that the annotator consider all the words explicitly. An annotator can move to the next pairs of sentences when all the words in the current pair are red, i.e. paired. An annotate pair is automatically saved when the annotator moves on to the next pair of sentences.

Besides providing the word-to-word similarity information, annotators were asked to judge whether the pair of sentences are indeed paraphrases or not. We wanted to compare such independent judgments with the original judgments provided by the MSRP designers. The annotation tool has a check button above the Prev and Next buttons at the bottom right corner of the SIMILAT's window that allows the annotators to specify whether they consider the two sentences to be in a paraphrase relation or not.

6. The SIMILAR Corpus

As we mentioned before, we selected a subset of the Microsoft Research Paraphrase (MSRP) corpus (Dolan, Quirk, and Brockett, 2004) to annotate. The MSR Paraphrase Corpus is the largest publicly available annotated paraphrase corpus which has been used in most of the recent studies that addressed the problem of paraphrase identification. The corpus consists of 5801 sentence pairs collected from newswire articles, 3900 of which were labelled as paraphrases by human annotators. The whole set is divided into a training subset (4076 sentences of which 2753 are true paraphrases) which we have used to determine the optimum threshold T, and a test subset (1725 pairs of which 1147 are true paraphrases) that is used to report the performance results.

There are several critiques about MSR corpus. First, MSR has too much word overlap (spawning from the way they collected the data set) and less syntactic diversity. Therefore, the corpus cannot be used to learn paraphrase syntactic patterns (Zhang and Patrick 2005; Weeds 2005). It should be noted that the lexical overlap is recognized by the creators of the corpus (Dolan and Brockett 2005) which indicate a .70 measure of overlap (of an unspecified form). The T-F split in both training and testing is quite similar though (67-33%).

Second, the annotations by humans were made on slightly modified sentences which are different from the original sentences publicly released. For instance, humans were asked to ignore all numbers and simply replace them with a generic token, e.g. MONEY for monetary values, and make judgments accordingly. This discrepancy between what humans used and what systems take as input complicates the task as some decisions are counterintuitive. For instance, the pair below was judged as a paraphrase although the percentages as well as the indices (*Standard & Poor* versus *Nasdaq*) are quite different.

T1: The broader Standard & Poor's 500 Index .SPX gained 3 points, or 0.39 percent, at 924.

T2: The technology-laced Nasdaq Composite Index < :IXIC > rose 6 points, or 0.41 percent, to 1,498.

Nevertheless, the MSRP corpus is the largest available and most widely used.

We annotated 700 pairs of sentences from the MSRP corpus which consists of 29,771 tokens (words and punctuation) of which 26,120 are true words and 17,601 content words. The number of content words is important because most of the semantic similarity metrics we used to derive semantic similarity scores with in order to relate to the human annotations only work on content words or certain types of content words, e.g. only between nouns or between verbs. The 700 pairs are fairly balanced with respect to the original MSRP judgments, 49% (344/700) of the pairs are TRUE paraphrases. Our own judgments yielded 63% (442) TRUE paraphrases for an overall agreement rate between our annotations and the MSRP annotations (both TRUE and FALSE paraphrases) of 75.7%. We simply instructed our judges to use their own judgment with respect to whether the two sentences mean

	Close	Related	Context	World Knowledge
Resnick	0.718	0.465	0.348	0.340
Leacock-Chodorow	0.862	0.639	0.596	0.499
Jiang and Conrath	0.774	0.268	0.190	0.191
Path	0.757	0.358	0.298	0.222
Lin	0.893	0.588	0.506	0.446
Wu and Palmer	0.886	0.701	0.605	0.578
LSA	0.292	0.228	0.136	0.204

Table 3. Average scores for each type of relation and each word-to-word similarity metric (all greedily matched pairs of words were included; from Text 1 to Text 2 and from Text 2 to Text 1).

	Close	Related	Context	World Knowledge
Resnick	0.702	0.5	0.33	0.249
Leacock-Chodorow	0.844	0.678	0.571	0.439
Jiang and Conrath	0.735	0.314	0.17	0.163
Path	0.728	0.412	0.268	0.188
Lin	0.869	0.632	0.449	0.339
Wu and Palmer	0.871	0.733	0.601	0.495
LSA	0.278	0.217	0.127	0.132

Table 4. Average scores for each type of relation and each word-to-word similarity metric for the optimally matched pairs for words.

	Close	Related	Context	World Knowledge
Resnick	0.375 (334/890)	0.634 (788/1242)	0.544 (241/443)	0.617 (169/274)
Leacock-Chodorow	0.336 (299/890)	0.559 (694/1242)	0.372 (165/443)	0.529 (145/274)
Jiang and Conrath	0.384 (342/890)	0.597 (742/1242)	0.424 (188/443)	0.693 (190/274)
Path	0.336 (299/890)	0.559 (694/1242)	0.372 (165/443)	0.529 (145/274)
Lin	0.416 (370/890)	0.648 (805/1242)	0.535 (237/443)	0.748 (205/274)
Wu and Palmer	0.336 (299/890)	0.561 (697/1242)	0.379 (168/443)	0.529 (145/274)
LSA	0.334 (297/890)	0.553 (687/1242)	0.381 (169/443)	0.507 (139/274)

Table 5. Percentage and raw numbers in parenthesis of pairs of greedily matched words for which the word-to-word semantic similarity metrics could not provide a score indicating their limitation.

	Close	Related	Context	World Knowledge
Resnick	0.383 (151/394)	0.619 (234/378)	0.58 (138/238)	0.721 (49/68)
Leacock-Chodorow	0.33 (130/394)	0.548 (207/378)	0.45 (107/238)	0.647 (44/68)
Jiang and Conrath	0.376 (148/394)	0.579 (219/378)	0.542 (129/238)	0.809 (55/68)
Path	0.33 (130/394)	0.548 (207/378)	0.450 (107/238)	0.647 (44/68)
Lin	0.414 (163/394)	0.614 (232/378)	0.630 (150/238)	0.853 (58/68)
Wu and Palmer	0.33 (130/394)	0.55 (208/378)	0.454 (108/238)	0.647 (44/68)
LSA	0.322 (127/394)	0.532 (201/378)	0.471 (112/238)	0.515 (35/68)

Table 6. Percentage and raw numbers in parenthesis of pairs of optimally matched words for which the word-to-word semantic similarity metrics could not provide a score indicating their limitation.

the same thing or not. MSRP guidelines were more targeted, e.g. judges were asked to consider different numerical values as being equivalent while we left such instructions unspecified. These differences in guidelines may explain the disagreements besides the personal differences in the annotators' background.

We have annotated so far 700 pairs. The 700 pairs were annotated by 6 different judges each annotating an equal, separate subset. As of this writing, a second judge annotates the same subset and we will be able to report inter-judge agreement. On a trial exercise of 100 pairs, inter-judge reliability was 63% at individual relation level.

Our effort resulted in a total of 12,560 relations of which 8,346 were IDENTICAL matches, 2849 relations detected greedily (890 CLOSE relations, 1242 RELATED relations, 443 CONTEXT relations, 274 KNOWLEDGE relations) and 1966 words were unmatched (a NONE type of relation was assigned to these words). For the optimum annotation, 15,692 relations were identified of which 8,046 were IDENTICAL and the judges identified 1,078 relations (394 CLOSE relations, 378 RELATED relations, 238 CONTEXT relations, 68 KNOWLEDGE relations) and 4,306 words were non matched.

We compared the human annotations with results

obtained with the word-to-word semantic similarity measures in the WordNet Similarity library (Pedersen, Patwardhan, and Michelizzi, 2004) as well as using LSA (Landauer et al., 2007).

We used the following similarity measures implemented in the WordNet::Similarity package and described in Pedersen, Patwardhan, and Michelizzi (2004): LCH (Leacock & Chodorow, 1998), RESNIK (Resnik, 1995), JIANG and CONRATH (Jiang & Conrath, 1997), LIN (Lin, 1998), PATH (Pedersen, Patwardhan, and Michelizzi, 2004) and WUP (Wu & Palmer, 1994). The WordNet-based similarity metrics require words with senses (i.e. concepts in WordNet; Miller, 1995) as input. We have experimented with all combinations of senses. We also used LSA as a word-to-word similarity metric. The LSA vectors were derived from a large collection of texts (the TASA corpus; Zeno et al., 1995).

The results are summarized in Tables 3-6. To obtain the results we took all matched words by humans and computed word-to-word similarity scores with each of the word-to-word semantic similarity metrics (shown in the first column) .Table 3 presents the average scores for all the similar words matched by the human annotators per the type of qualitative similarity relation identified by the annotators. Table 3 presents results for similar words that were greedily matched while Table 4 for words optimally matched. Table 3 combined the results for the greedy annotations in both directions: matching target words from text A to words in text B and also matching target words from text B to words in text A. From both tables 3 and 4 we can clearly see that the averages for each type of relations are very different with few exceptions. For instance the Jiang and Conrath and the LSA cannot distinguish between CONTEXT and KNOWLEDGE types of relations when optimally matched. LSA yields very close averages for RELATED and KNOWLEDGE types of relations when greedily matched. Resnick also has problems separating the CONTEXT from KNOWLEDGE word matchings when greedily matched as the corresponding averages are very close.

When analyzing the results in Tables 5 and 6, which represent the percentages of pairs of words by annotators for which the word-to-word semantic similarity metrics could not provide a score (i.e. misses), we realized that LSA is the most robust as it has least misses. The other measures are constraint to only content words or only certain types of words, e.g. nouns or verbs. LSA could compute the similarity between a pronoun and noun, for instance, while any of the WordNet Similarity metrics cannot. The Lin measure yields the most misses.

7. Further Work

We plan to continue our work presented in this paper along several lines of future research. First, we would like to annotate more data to have a larger annotated corpus. Furthermore, we would like to add another level of annotation in which we indicate phrases that are semantically equivalent without the need to matched particular words within those phrases. Such examples of equivalent phrases which do not need to be decomposed further into word-level matchings are "giving birth" and "have an offspring". Second, we plan to use the greedily matched pairs and the optimally matched pairs by human annotators in automated methods and compare the results thus obtained with the fully greedy and automated methods. Finally, we would like to propose a qualitative model of word-level semantic similarity.

8. Conclusion

We have described in this paper a novel protocol to annotate texts with qualitative judgments of word-level similarity. A greedy and optimal annotation strategy was developed and implemented. The word-to-word annotations by human judges were related to quantitative scores of similarity generated by a set of WordNet-based similarity metrics and LSA. The comparison revealed the strengths and weaknesses of these metrics which in turn has important implications for future developments of text-to-text similarity methods and other methods that will include the word-to-word similarity metrics.

9. Acknowledgements

This research was supported in part by HIDDEN. Any opinions, findings, and conclusions or recommendations expressed in this material are solely the authors' and do not necessarily reflect the views of the sponsoring agency.

10. References

Blei, D. M., Ng, A. Y., and Jordan, M. I. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* **3** (4–5): *pp.* 993–1022.

Corley, C. and Mihalcea, R. 2005. Measures of Text Semantic Similarity, in *Proceedings of the ACL workshop on Empirical Modeling of Semantic Equivalence*, Ann Arbor, MI, June 2005

Dagan, I., Glickman, O., and Magnini, B. 2005. The PASCAL recognizing textual entailment challenge. In *Proceedings of the PASCAL Workshop*.

Dasgupta, D., Niño, F., Garrett, D., Chaudhuri, K., Medapati, S., Kaushal, A., Simien, J. 2009. A multi-objective evolutionary algorithm for the task based sailor assignment problem. GECCO 2009: 1475-1482.

Dawes, M. 2011. *The Optimal Assignment Problem*, Course notes, University of Western Ontario. (accessed online in December 2011)

Dolan, W.B., Quirk, C., and Brockett, C. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th International Conference on Computational Linguistics*, Geneva, Switzerland.

Graesser, A.; Hu, X.; and McNamara, D. 2005. Computerized learning environments that incoporate research in discourse psychology, cognitive science, and computational linguistics. In Healy, A., ed., *Experimental Cognitive Psychology and its Applications*, 59–72. Washington, D.C. American

Psychological Association.

Graesser, A.; Olney, A.; Hayes, B. C.; and Chipman, P. 2005. Autotutor: A cognitive system that simulates a tutor that facilitates learning through mixed-initiative dialogue. In *Cognitive Systems: Human Cognitive Models in System Design*. Mahwah: Erlbaum.

HIDDEN HIDDEN HIDDEN HIDDEN HIDDEN HIDDEN

Ibrahim, A., Katz, B., and Lin, J. 2003. Extracting structural paraphrases from aligned monolingual corpora. In *Proceeding of the Second International Workshop on Paraphrasing*, (ACL 2003).

Iordanskaja, L., Kittredge, R., and Polgere, A. 1991. Natural Language Generation in Artificial Intelligence and Computational Linguistics. Lexical selection and paraphrase in a meaning-text generation model, Kluwer Academic.

Jiang, J.J. & Conrath, D.W. 1997. Semantic Similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the International Conference on Research in Computational Linguistics*.

Kuhn, H.W. 1955. "The Hungarian Method for the assignment problem", *Naval Research Logistics Quarterly*, 2:83–97, 1955. Kuhn's original publication.

Landauer, T.K..; McNamara, D.S.; Dennis, S.; and Kintsch, W. 2007. *Handbook of Latent Semantic Analysis*. Mahwah, NJ: Erlbaum.

Leacock, C.; and Chodorow, M. 1998. Chapter: Combining local context and WordNet sense similarity for word sense identification. *WordNet, An Electronic Lexical Database*. The MIT Press.

Lintean, M., & Rus, V. (2009). Paraphrase Identification Using Weighted Dependencies and Word Semantics. Proceedings of the 22st International Florida Artificial Intelligence Research Society Conference. Sanibel Island, FL.

Lintean, M., Moldovan, C., Rus, V., & McNamara D. (2010). The Role of Local and Global Weighting in Assessing the Semantic Similarity of Texts Using Latent Semantic Analysis. Proceedings of the 23rd International Florida Artificial Intelligence Research Society Conference. Daytona Beach, FL.

McCarthy, P.M. and McNamara, D.S. 2008. User-Language Paraphrase Corpus Challenge, online, 2008.

Miller, G. 1995. WordNet: A Lexical Database of English. *Communications of the ACM*, v.38 n.11, p.39-41.

Pedersen, T.; Patwardhan, S.; and Michelizzi, J. 2004. WordNet::Similarity – Measuring the Relatedness of Concepts. In *Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2004.*

Resnik, P. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*.

Rus, V. & Graesser, A.C. 2006. Deeper Natural Language Processing for Evaluating Student Answers in Intelligent Tutoring Systems, *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI-06)*.

Rus, V., McCarthy, P. M., Lintean, M., McNamara, D. S., and Graesser, A. C. 2008. Paraphrase Identification with Lexico-Syntactic Graph Subsumption. In *Proceedings of the Florida Artificial Intelligence Research Society International Conference (FLAIRS-2008)*.

Rus, V., Lintean, M., Azevedo, R. (2009). Automatic Detection of Student Mental Models During Prior Knowledge Activation in MetaTutor. Proceedings of the 2nd International Conference on Educational Data Mining. Cordoba, Spain.

Rus, V., Nan, X., Shiva, S., & Chen, Y. 2009. Clustering of Defect Reports Using Graph Partitioning Algorithms, *Proceedings of the 20th International Conference on Software and Knowledge Engineering*, July 2-4, 2009, Boston, MA. Rus, V., McCarthy, P. M., Lintean, M., McNamara, D. S., and Graesser, A. C. 2008. Paraphrase Identification with Lexico-Syntactic Graph Subsumption. In *Proceedings of the Florida Artificial Intelligence Research Society International Conference (FLAIRS-2008)*.

Rus, V. and Lintean (2012). A Comparison of Greedy and Optimal Assessment of Natural Language Student Input Using Word-to-Word Similarity Metrics. Proceedings of the International Conference on Intelligent Tutoring Systems. Crete, Greece.

Salton, A. G., Wong, and C. S. Yang. 1975. "A Vector Space Model for Automatic Indexing," *Communications of the ACM*, vol. 18, nr. 11, pages 613–620.

VanLehn, K., Graesser, A. C., Jackson, G. T., Jordan, P., Olney, A. M., & Rose, C. 2007. When are tutorial dialogues more effective than reading? *Cognitive Science, 31*, 3-62.

Weeds, J., Weir, D., & Keller, B. 2005. The distributional similarity of sub-parses. In Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pages 7–12, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Wu, Z.; and Palmer, M.S. 1994. Verb Semantics and Lexical Selection. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Zeno, S., Ivens, S., Millard, R., & Duvvuri, R. 1995. *The educator's word frequency guide*. Brewster, NY: Touchstone Applied Science Associates.

Zhang, Y. & Patrick, J. 2005. Paraphrase identification by text canonicalization. In Proceedings of the Australasian Language Technology Workshop.

Using Semantic Relations to Solve Event Coreference in Text

Agata Cybulska, Piek Vossen
VU University Amsterdam
De Boelelaan 1105
1081HV Amsterdam
E-mail: a.k.cybulska@vu.nl, piek.vossen@vu.nl

Abstract

In this paper, we report on how semantic relations between event mentions in text can be used to solve event coreference. Event descriptions in text differ in specificity and granularity. We believe that based on meronymy and hyponymy relations between event mentions one can determine shifts in levels of granularity and abstraction and use these as indication for coreference resolution. This article presents a model that captures the relationship between semantic relations amongst events and event coreference. A number of heuristics can be used to estimate semantic distance between instances of event descriptions and based on that to calculate coreference match between event mentions. Within this study we used the Leacock-Chodorow similarity measure as a heuristic for event coreference resolution. We report about the success rates of our experiments based on the evaluation performed on a corpus annotated with coreferent events.

Keywords: event coreference, semantic relations

1. Introduction

The research project "Semantics of History" [1] is concerned with the development of a historical ontology and a lexicon that will be used in a new type of information retrieval system which can handle the time-based dynamics and varying perspectives in historical archives. Historical texts focus on changes in reality that happen over time (Ide & Woolner, 2007). Historical realities can be seen differently depending on the subjective view of the writer. In the design of our search system, the change of reality and the diverse attitudes of writers towards historical events will be considered and both will be used for the purpose of historical information retrieval.

In the first phase of the project we did research on how descriptions of historical events are realized in different types of text and what the implications are for historical information retrieval. Texts, written shortly after an event happened, use more specific and uniquely occurring event descriptions than texts describing the same events but written from a longer time perspective[2]. To capture differences between event representations and to identify relations between historical events, for the purpose of our work we applied a historical event model which consists of 4 slots: a location slot, time, participant and an action slot (see also Van Hage et al 2011 for the formal SEM model along the same lines). After arriving at an understanding of how to model historical events, we moved on to extracting events from text[3]. We extracted conflict related event actions (which are the focus of this project) and their participants, locations and time markers from text within the KYOTO framework; based on some syntactic clues, PoS, lemma and combinatory information together with semantic class definition and exclusion by means of WordNet. Having extracted instances of events and event participants we moved on to determining relations between event mentions, starting with coreference resolution.

Event descriptions in text differ in specificity and granularity. High level events, such as *war*, are more general and abstract with longer time span and group participants; low level events, for instance a *shooting* event, are rather specific with shorter duration, and individual participants. Based on meronymy and hyponymy relations between event mentions one can determine shifts in levels of granularity and abstraction and use these as a clue for event coreference and identification of other event relations. Within the Semantics of History research project we are interested in solving event coreference as well the identification of event relations such as sub-event, causal and temporal relations. But the focus of this study lies on the identification of semantic relations between event descriptions for the purpose of solving event coreference.

In accordance with the Quinean theory (1985), we claim that semantic relations and coreference between elements of the contextual setting[4] of events are crucial for solving event coreference. In the case of conflict–related events, such as: *war, genocide, bombings, shootings, killings, fighting, aggression*, the contextual setting of the event as well as its participants cannot be separated from the event itself, i.e. they constitute the event. Time and place in which an event happened form the starting point for solving event coreference (compare: *genocide in*

[1] The Semantics of History is funded by the Interfaculty Research Institute CAMeRA at the Free University Amsterdam as a collaboration of the Faculties of Arts and Exact Science: http://www2.let.vu.nl/oz/cltl/semhis/index.html.

[2] For details see Cybulska, Vossen, 2010.

[3] See Cybulska, Vossen, 2011.

[4] By contextual setting we mean the time when and place where an event happened.

Srebrenica with *genocide in Rwanda*). For each event mention in text, one should ideally first try to define the time and place and after that, for events occurring within a particular time frame and space, search further for coreference clues[5].

Our approach determines whether two events corefer based on the combination of coreference scores calculated for event components: event participants, location, event time and action. Traditional approaches to solving event coreference tend to concentrate on the actual events only (event actions as we call them); we rather see event actions as just one piece of the puzzle, which alone is not enough to determine relations between historical events described in text (compare: *two students taken hostage in Beslanian school* vs. *two people taken hostage in a classroom in Beslan Russia*). In fact, we think that coreference is not an absolute notion. For example, *shooting* and *several shots* can more or less refer to the same event and people may have different or vague intuitions about their identity. A gradable notion of coreference is therefore both operational (for robust automatic detection) and possibly psychologically adequate. That is why for each event pair in the text, we want to calculate a coreference match score as a combination of coreference scores collected for pairs of event components. To obtain the match score for an event component, we will analyze semantic relations and semantic distance between two instances (for instance participants of event A in comparison with participants of event B). We believe that shifts vs. agreement in the level of granularity and in the level of abstraction will play a crucial role in the assignment of the match scores; obviously together with other coreference indicators such as identification of repetition, anaphora, synonymy and disjunction. The cumulative coreference match score gathered by an event pair will indicate whether two events can be considered likely candidates for exhibiting a coreference relation.

[5] Determining event time and place information should considerably limit the number of candidates for coreferent events. In practice it often happens that the time and place information of an event is not available, but it sure would be a waste not to make use of this information whenever it can be found in the text or learned from other knowledge sources.

In this paper, we report on a study on how semantic relations between event mentions in text can be used to solve event coreference. After the introductory section 1, in section 2 we describe related work with regards to event coreference resolution and application of semantic shifts in NLP applications. In chapter 3 we propose a model capturing the relationship between semantic relations and coreference resolution. In chapter 4 we describe heuristics used to evaluate the model; in section 5 we present some preliminary evaluation results and in final chapter 6 we draw conclusions.

2. Related Work

Using semantic shifts in NLP applications is not a new idea. Mulkar-Mehta, Hobbs and Hovy (2011) investigated granularity shifts and granularity structures in natural language text. They focused on modeling part-whole relations between entities and events and causal relations between coarse and fine granularities. In their follow-up work, they described an algorithm for extracting causal granularity structures from text and its possible applications in question answering and text summarization. In our work, we want to use shifts in granularity but also in abstraction for the purpose of event coreference resolution. To the best of our knowledge, identification of semantic shifts has not been used before for this task.

Interesting approaches to coreference resolution between event actions in text have been proposed by Bejan and Harabagiu (2010) and Chen et. al (2011). Bejan and Harabagiu (2010) experimented with solving coreference between event actions by means of two nonparametric Bayesian models employing a combination of lexical and class features (such as PoS and semantic classes of events) together with WordNet features (WordNet synonyms and supersenses) and predicate – argument structures[6]. Solving within document coreference on the ACE data set (restricted set of event types as LIFE, BUSINESS, CONFLICT, JUSTICE) they achieved the highest performance results of 83.8% B^3 F-measure (B^3 metric by Bagga and Baldwin, 1998) while on their newly created EventCorefBank (ECB corpus with articles on 43 different topics from the GoogleNews archive) they reached ca. 90% B^3 F-score. Their approach does not account for partial coreference of events, where some of the event components are related through hyponymy and/or part-of relationship, which is the focus of our work (noted by the authors as the reason for one of the common errors in their output).

The same holds for Chen et. al (2011) who propose a two step framework for resolution of coreference between event actions and their objects. To identify coreferent mention pairs they employ support vector machine with

[6] For details see Bejan, Harabagiu 2010.

tree kernels. Seven distinct mention pair resolvers are using a combination of lexical, PoS, semantic and syntactic features (amongst others an argument matching feature to account for different syntactic structures and a semantic type feature with types such as *person*, *location* etc). Then, to form coreference chains spectral graph partitioning is used. Within-document-coreference is solved between nominal, verbal and pronominal descriptions of events and objects with 46.91% B^3 F-score on the OntoNotes 2.0 corpus, annotated with coreference between all event mentions (not using any pre-defined concept types as in the ACE corpus). This approach accounts for synonymy relation between mentions but neither for meronymy nor hyponymy relations.

3. Proposed Model: Semantic Relations and Coreference Resolution

To capture differences between event representations, we applied an event model which consists of 4 components: action, participant, location, and time. In textual data one comes across specific and general actions, participants, time expressions and locations[7]; compare for instance event actions such as *shooting*, *fighting, genocide* and *war*, or participants: *soldier* versus (multiple) *soldiers* vs. *troops* and *multiple troops*; the same holds for time markers as *day, week* and *year* and also for event locations: *city* vs. *region* vs. *continent*. Event mentions are either (partially) overlapping or disjoint.

Sematic Rels / Event components	Meronymy *Part-of, membership*	Is-a *Class>Subclass*	Instance-of *Class>Instance*
Location	*Bosnia>Srebrenica*	*city>capital*	*city>Srebrenica*
Participants	*army>soldier*	*officer>colonel*	*colonel>Karremans*
Time	*week>Monday*	*weekday>Friday*	*year>1995*
Action	*series of attacks> 1 attack*	*attack>bombing*	*genocide> Srebrenica massacre*

Figure 1: Relations between general and specific event mentions

Next to rather clear indicators that are typically used in coreference resolution such as repetition, synonymy, anaphora and disjunction (negative indicator) [8], significant relations between event components are along a hyponymy axis: class vs. its subclass such as *officer* being a subclass of the class *person*, instance-of a class such as *Bosnia* being an instance of the class *country*; and along a meronymy axis: member vs. group i.e. *Colonel Karremans* being a member of the group of *Dutch UN soldiers* or part vs. whole relation such as *Srebrenica* being a part of *Bosnia*.

On top of different degrees of granularity and abstraction, words and word combinations on the same level of granularity and abstraction may differ in terms of pragmatic use, while potentially referring to one and the same thing; compare for example event participants referred to as *aggressors* and *liberators* or *troops*, *army* and *soldiers*. The same applies to event actions; compare *liberation* with *invasion* or *military intervention*. When solving event coreference and determining event relations, the pragmatic loading has to be accounted for as well. In other words, one has to be able to distinguish between subjective marking and proper semantic disjunction.

Our hypothesis is that abstraction and granularity agreement between complete events can be determined by the semantic relations between the event components (below referred to as *Ec*). We thus first define per event component a coreference match (below referred to as *CM*) as a function of the relation type and semantic distance between the instances of components. The highest coreference match (value 1) will be assigned to synonymous items, repetitions, as well as to anaphora in case their number and gender agree:

$$CMrepetition(Ec1, Ec2) = 1$$
$$CManaphora(Ec1, Ec2) = 1$$
$$CMsynonymy(Ec1, Ec2) = 1$$

Similarly, a high match score is used for events with only a difference in perspective (for instance *buy* vs. *sell*):

$$CMperspective(Ec1, Ec2) = 1$$

We further expect that hyponymy relations across event components indicate a probability of coreference. Our formula expresses that distance inversely correlates with the likelihood of coreference:

$$CMhyponymy(Ec1,Ec2) = \frac{1}{(1 + |\Delta (Eh_l(Ec1), Eh_l(Ec2))|)}$$

where Eh_l stands for the estimated hyponymy level within a shared chain of hyponymy relations for *Ec1* and *Ec2* in a resource such as WordNet. By shared we mean that the concepts are not disjoint according to the interpretation of the hierarchy (see for instance the hyponymy chain from English WordNet connecting the concepts of hostage and person: *hostage<captive/prisoner<unfortunate<person <being/organism<living thing*).

Meronymy relations between instances are expected to indicate granularity shifts, where the value of *CM* inversely correlates with the difference in size of the

[7] Cybulska, Vossen, 2010.

[8] The identification of disjunction as well as of synonymy relations, repetitions and anaphora is obviously of importance for this task, but we will not discuss it further in this paper.

meronymic whole, i.e. the larger the difference in size, the lower the score. This is formalized as follows:

$$\text{CMmeronymy}(Ec1,Ec2) = \frac{1}{(1 + |\Delta (En_i(Ec1), En_i(Ec2))|)}$$

where $En_i(Ec)$ stands for the estimated number of individuals denoted by *Ec*. En_i can be based for instance on predefined levels of granularity, where a large difference in levels correlates with a large difference in the number of denoted individuals. The following levels of concepts per event component will be distinguished based on a knowledge base:

- for participants: *person* and *group*
- for locations: (up to a) *building* level vs. *city* vs. *country* level
- for time expressions: *hourly* level (less than a day), *day* level, *week*, *month*, *year*.

Obviously, one must consider multiplications within a level as well: 24 hours make 1 day.

If two instances are disjoint (for instance human participants of different gender) the match score will equal zero:

$$\text{CMdisjunction}(Ec1, Ec2) = 0$$

Once the above values have been calculated for every component of an event pair, the collected scores will be combined into a single score for an event pair indicating the likelihood of coreference. Our model predicts that, except for the clear cases resulting in an absolute score of 1 or 0, event components that are far apart in terms of meronymy and hyponymy have an extreme difference in granularity and abstraction and therefore a low likelihood to establish coreference. A participant example would be a *US sergeant* (specific in terms of hyponymy and a single-form) versus *human being*, where the latter does not exclude *US sergeants* but there is a low likelihood that we are talking about the same thing. For events, this could be a *briefing by an US sergeant* versus *strategics*. Through empirical testing, we can then determine thresholds for establishing optimal coreference relations across events (components).

Within events we observe granularity and abstraction correlations. If an event action is rather abstract and general (for instance *war*) one can expect the participants of this action to be a multiform and certainly not a single individual. The same holds for the location of a *war* event (also generic such a territory of a country instead of a small area) and its time span (a longer time period). This observation offers a perspective that it may be possible to determine the granularity and abstraction level of one event component from those of other components with which it often co-occurs.

In the ideal situation, one has information on all event components. More realistic is the situation where event components are underspecified in the event mentions, for instance in the case of nominalizations (*war, shooting*). Underspecified nominalizations (no time, place and no event participants made explicit) tend to refer in a more general and abstract way to events that are expected to be described earlier in the text in more detail and so on a lower generosity level. Incomplete events will be analyzed in a separate way. An interesting possibility is to try to learn the missing event information from other (knowledge) sources (for instance in case of named events from Wikipedia).

4. Evaluation of the Model - Experiments

Different heuristics can be employed to estimate semantic distance between event mentions in text. Two groups of techniques can be distinguished that can be used to define the difference in hyponymy and meronymy: (1) analysis of the text and of the morpho-syntactic properties of event mentions and (2) using background knowledge: either learned from existing resources as WordNet, geo- and temporal ontologies or knowledge mined based on probability estimates from the internet corpus. Regarding the latter, one could for instance try to learn the typical length of duration that is most frequently associated with an action and use this for abstraction and meronymy estimates.

In this section we report on experiments that were performed to determine semantic distance between event actions based on the distance in the WordNet database. Leacock and Chodorow similarity measure (1998) was used where next to the path length in WordNet also the relative depth in the knowledge base is considered[9].

For the experiments we used the gold standard set of 66 texts[10] from the Intelligence Community (IC) Corpus that were annotated (amongst other relations) with within document coreference between violent events as *bombings, killings, wars* etc.; belonging to an event ontology of ca. 50 terms (Hovy et al. 2012). The corpus was created at the Information Sciences Institute of the University of Southern California within the context of a project on automated deep reading of text (Chalupsky et al., 2012).

The 66 manually annotated texts were processed by means of tools developed within the KYOTO project[11]. First, the corpus was lemmatized; and tagged with PoS-information. Next, word sense disambiguation was

[9] In the future we want to experiment with other methods to define semantic similarity.

[10] The annotation of the IC Corpus is an ongoing process. At the time when this research was performed the gold standard consisted of 66 texts.

[11] KYOTO tools are a pipeline-architecture of linguistic processors that were specifically designed to extract events with their participants from text. For more information on the KYOTO project go to http://www.kyoto-project.eu/.

performed and the corpus was semantically annotated with synsets from the English Wordnet and with predefined ontological classes[12]. In the IC Corpus all violent event actions (also the coreferent ones but not only) were manually annotated. All annotated event actions from the corpus were used as input in the experiments. To extract participants of event actions a newly created participant extraction module for English was created based on manual annotation of participants in 5 texts from the IC Corpus. By means of the Kybot module of KYOTO architecture event participants were extracted based on some syntactic clues, PoS, lemma and combinatory information together with semantic class definition and exclusion by means of Wordnet. In the future the same procedure will be applied for the extraction of event time and locations.

To generate candidates of coreferent mentions semantic distance was calculated between heads of all action phrases (verbal, nominal, pronominal, elliptic, etc.) that were automatically extracted by means of KYOTO tools. The KYOTO system outputs the highest scoring WordNet synset for each head that lead to the match. The matches are based on an event ontology that was manually assigned to hypernyms in WordNet. Next, the Leacock and Chodorow measure was used to identify chains of mentions with the shortest semantic distance and thus potential coreference chains. The measurement considers the closest hyponymy path in WordNet between two synsets scaled by the overall depth of the taxonomy. We calculated the overall depth for all the event mentions in the document rather than using a single overall measure based on WordNet. A special case is formed by mentions that use the same word. In that case, we ignore the synset assigned but consider a distance of 1. If different words are synonyms, we use a distance score of 2. For all other cases, we add the hypernym distance to the initial value of 2. Following Leacock and Chodorow, we calculate the similarity using the formula:

log(distance/(2*averageDepth)

We created a matrix between all mentions in a document and calculated the Leacock and Chodorow similarity scores. From this matrix, we determined the optimal coreference thresholds - for actions from 0.50 similarity measure upwards and for participant mentions from 0.51 of the Leacock and Chodorow score.

In our future work, we will use the coreference scores of the participants of related events to further fine-tune coreference scores between the actions they participate in (the same goes for event time markings and event locations).

[12] For details on our approach to extract events and participants using the historical ontology see Cybulska, Vossen, 2011.

5. Evaluation Results

In the evaluation phase the manual annotations of coreferent actions from the IC Corpus were used as key chains and were compared with the response chains generated by means of the above described heuristic. For comparison we also show evaluation results for event participants (at this point no key chains were used for the evaluation of participants). Since our goal was to evaluate the importance of hyponymy relations for the task of coreference resolution, we used a baseline that assigns a coreference relation to all nouns and verbs that belong to the same lemma (*True Baseline*). This baseline does not consider whether a word refers to an event, participant or any other textual element. We also added a more specific baseline (*Lemma*) that checks the lemma overlap for just the events and participants (participants extracted through the KYOTO system). Using this baseline, we can measure the contribution of the WordNet hierarchy in our approach in addition to matching just lemmas for the extracted events and participants.

Table 1 presents coreference evaluation results achieved by means of the Leacock and Chodorow similarity measure (*L&C*) as heuristic in comparison to the word match baseline results in terms of recall (*R*), precision (*P*) and F-score (*F*) by employing the commonly used coreference resolution evaluation metrics B^3 (Bagga and Baldwin, 1998). Considered that the presence of singletons in the gold standard and in the system output (and especially in the gold standard) artificially boosts the evaluation scores (Kuebler and Zhekova, 2011) we also present the evaluation results in *B^3-singletons* - after the removal of singletons from both the key chains and the response chains.

Event Slot	Heuristic	B^3			B^3-singletons		
		R	P	F	R	P	F
All Actions	Lemma	0.80	0.78	0.79	0.58	0.68	0.62
	L&C	0.81	0.71	0.76	0.60	0.61	0.60
Event Part.	Lemma	0.71	0.79	0.74	0.38	0.58	0.46
	L&C	0.70	0.72	0.71	0.38	0.52	0.44
All N&V	True Baseline	0.80	0.68	0.73	0.58	0.41	0.48

Table 1: Event Coreference Evaluation Results in B^3 metrics and in B^3 after removal of singletons (micro averages)

Compared to the *True Baseline* (considering the lemmas of *all* nouns and verbs) our coreference resolution performs equal in recall and much better in precision. Our F-measure scores 3% higher for B^3 and 12% higher on B^3 without considering singletons references. This is a significant difference.

When we compare the actions resolved through *L&C* with the lemma baseline on just the extracted actions (*Lemma*), we see that out approach adds a little bit recall

but looses a lot on precision. The F-measures are lower. This holds both for actions and participants. This is what we expect, since this baseline is a very precise but conservative approach. There can be 3 reasons for the low precision:

1. We selected the wrong synsets for the words;
2. The wordnet hierarchy and/or the Leacock-Chodorow similarity does not properly reflect true similarity;
3. Other relations than hyponymy play a role.

A further error analysis needs to reveal how these factors play a role and how we can improve the results.

Compared to evaluation results achieved in related work (Bejan and Harabagiu 2010 – 83.8% B^3 F-score and Chen et. al 2011 – 46.91% B^3 F-score) by means of our approach coreference between event actions was solved with a relatively high 76% B^3 F-score, especially considering that for coreference resolution exclusively a simple heuristic based on WordNet distance was used.
The F-score of 60% achieved in B^3 measure but after removal of singletons seems to be a more realistic performance estimate but still demonstrating the significance of partial coreference resolution between mentions related through hyponymy for the coreference resolution task.

For the sake of comparison, we also present the evaluation results of participant coreference resolution (evaluation without a key chain) resulting in a B^3 F-scores of 71% and 44%. These results are still better than the *True Baseline*, which also does not differentiate between events and participants. We do see that recall for our approach (both using *L&C* and using *Lemma*) is considerably lower. This makes sense, since the gold-standard was not intended for participants and the *True Baseline* simply extracts all.

6. Conclusion and Future Work

In this paper, we presented a model to capture the relationship between semantic relations and coreference resolution. Our preliminary evaluation results showed that semantic relations can be used successfully for the purpose of coreference resolution. Especially the importance of hyponymy relations in resolution of coreference was demonstrated in our experiment with a simple heuristic employing semantic distance measurement as the only coreference indication and achieving comparatively good evaluation results. If combined with other coreference features a significant improvement of performance is to be expected.

In the future we will further test our event coreference resolution model by using other heuristics to find coreference candidates and by applying these heuristics to all event components: besides actions to participants, locations and event time markings. As the next step in our work, heuristics will be employed that make use of meronymy - the part-of and member relations between event components (amongst others through usage of granularity ontologies) and combine these with WordNet distance techniques.
Also, experiments will be performed on how to best cumulate the general coreference match score as a combination of coreference matches of all event components.

7. Acknowledgements

This research was funded by interfaculty research institute CAMeRA (Center for Advanced Media Research) at the Free University Amsterdam (http://camera.vu.nl). Part of this work was completed at the Information Sciences Institute (ISI) of the University of Southern California under the auspices of Prof. Eduard Hovy. We are grateful for Prof Hovy's collaboration in allowing us to perform the evaluation on the Intelligence Community Corpus annotated at ISI.

8. References

Bagga, Amit and Breck Baldwin, "Algorithms for Scoring Coreference Chains", in *Proceedings of the 1st International Conference on Language Resources and Evaluation*, LREC 1998

Chalupsky, Hans et al., "The RACR Machine Reading System", in prep. 2012

Chen, Bin, Su, Jian, Pan, Sinno Jialin and Chew Lim Tan, "A Unified Event Coreference Resolution by Integrating Multiple Resolvers", in *Proceedings of the 5th International Joint Conference on Natural Language Processing*, p. 102-110, Chiang Mai, Thailand, November 8 – 13, 2011

Cosmin Adrian Bejan, Sanda Harabagiu, "Unsupervised Event Coreference Resolution with Rich Linguistic Features", in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics,* Uppsala, Sweden, 2010

Cybulska, Agata and Piek Vossen, "Event models for Historical Perspectives: Determining Relations between High and Low Level Events in Text, Based on the Classification of Time, Location and Participants", in *Proceedings of LREC 2010*, Valletta, Malta, May 17-23, 2010

Cybulska, Agata and Piek Vossen, "Historical Event Extraction from Text", in *Proceedings of ACL LaTeCH*, Portland, US, June 2011

Hovy, Eduard, Mitamura, Teruko, Verdejo, Felisa, Philpot, Andrew and Agata Cybulska, "Identity and Quasi-Identity Relations for Event Coreference", in prep. 2012

Kuebler, Sandra and Denislava Zhekova, "Singletons and Coreference Resolution Evaluation", in *Proceedings of Recent Advances in Natural Language Processing,* p. 261-267, Hissar, Bulgaria, September 12-14, 2011

Leacock, Claudia and Martin Chodorow, "Combining local context with WordNet similarity for word sense

identification", in Christiane Fellbaum (ed.), *WordNet : A lexical Reference System and its Application*, MIT Press, Cambridge, MA.

Mulkar-Mehta, Rutu, Hobbs, Jerry R. and Eduard Hovy, Applications and Discovery of Granularity Structures in Natural Language Discourse, in *Proceedings of The Tenth International Symposium on Logical Formalizations of Commonsense Reasoning at the AAAI Spring Symposium*, Palo Alto, 2011

Mulkar-Mehta, Rutu, Hobbs, Jerry R. and Eduard Hovy, "Granularity in Natural Language Discourse", in *Proceedings of International Conference on Computational Semantics*, 2011

Quine, Willard V., "*Events and Reification*" in E. LePore and B.P. McLaughlin (eds.), *Action and Events*, Basil Blackwell, New York, 1985

Van Hage, Willem, Malaisé, Veronique, Segers, Roxane, Hollink, Laura (fc), Design and use of the Simple Event Model (SEM), the *Journal of Web Semantics*, Elsevier

Using semantic resources to improve a syntactic dependency parser

Gerold Schneider

Institue of Computational Linguistics, University of Zurich
gschneid@cl.uzh.ch

Abstract

Probabilistic syntactic parsing has made rapid progress, but is reaching a performance ceiling. More semantic resources need to be included. We exploit a number of semantic resources to improve parsing accuracy of a dependency parser. We compare semantic lexica on this task, then we extend the back-off chain by punishing underspecified decisions. Further, a simple distributional semantics approach is tested. Selectional restrictions are employed to boost interpretations that are semantically plausible. We also show that self-training can improve parsing even without needing a re-ranker, as we can rely on a sufficiently good estimation of parsing accuracy. Parsing large amounts of data and using it in self-training allows us to learn world knowledge from the distribution of syntactic relation. We show that the performance of the parser considerably improves due to our extensions.

Keywords: Exploitation of semantic resources for NLP applications, Syntactic parsing, WordNet and WordNet-like resources, Self-training, Distributional semantics

1. Introduction

Syntactic parsing has made impressive progress over the past decade. Still, performance even of the best parsers lags behind human performance considerably. Bi-lexical statistics (Collins, 1999) has led to a quantum leap in parsing performance. The interaction of lexis and grammar, as postulated by (Sinclair, 1991) or (Hunston and Francis, 2000), is exploited by bi-lexical statistics for the disambiguation task. In terms of psycholinguistics, prefabricated partial trees are recognized directly and usually not decomposed into subparts. In terms of semantics, lexical semantics is modeled as the distribution of grammatical relations between lexemes at the syntactic level and can be used to discover similar words (Lin, 1998) or WordNet synsets (Curran, 2004). (Grefenstette et al., 2011) present a compositional distributional model of meaning in vector space models (e.g. (Schütze, 1998)), where the semantic vector space of a word is defined in terms of its distributional syntax.

The performance of statistical parsers is now reaching a ceiling. Additional types of semantic resources need to be considered and included. We present experiments using an existing dependency parser and investigate the role of semantics for parser improvement in this paper. Two semantic lexica are compared for the reduction of data sparseness. We extend the backoff chain by punishing underspecified decisions. Further, a simple distributional semantics extension is tested. We then use selectional restrictions to boost interpretations that are semantically plausible. We also show that self-training can improve parsing even without using a re-ranker. Parsing large amounts of data and using it in self-training allows us to learn world knowledge from the distribution of syntactic relation.

1.1. The Pro3Gres parser

The parser used in this study, Pro3Gres (Schneider, 2008), is a Dependency parser. Its representation is very close to and can be mapped to GREVAL (Carroll et al., 2003) and the Stanford scheme (Haverinen et al., 2008).

The parser uses a hand-written *competence* grammar and a statistical *performance* disambiguation learnt from the Penn Treebank (Marcus et al., 1993). The parser uses a Maximum Likelihood Estimation (MLE) probability model for the bi-lexical performance disambiguation, which we briefly introduce here in preparation for the adaptations that we make in the paper. The parser estimates the probability of the dependency relation R at distance (in chunks) $dist$, given the lexical head a of the governor and the lexical head b of the dependent.

$$p(R, dist|a, b) = P(R|a, b) \cdot P(dist|R, a, b) \quad (1)$$

$$\cong \frac{\#(R, a, b)}{\#((\sum R), a, b)} \cdot \frac{\#(R, dist)}{\#R} \quad (2)$$

The assumption is taken that the distance depends only on the relation type, and that a relation is only ambiguous in terms of the relations with which it is in competition. In order to alleviate sparse data, the parser uses a back-off architecture similar to (Collins and Brooks, 1995), but it extends from PP-attachment to most of its dependency relations, and includes simple semantic classes from WordNet (Miller et al., 1990), as e.g. in (Merlo and Esteve Ferrer, 2006).

The MLE probability model and the backoffs differ slightly for some relations. We now describe the PP-attachment model, which uses tri-lexical disambiguation. PP-attachment is modeled as ambiguous between noun attachment and verb attachment (the latter including adjective attachment). It uses the putative parsing context of (Collins and Brooks, 1995) as an approximation, where every verb is in competition with one noun, and every noun is in competition with one verb. The actual competitions during parse time are never in direct comparison, but indirectly via the comparison of the putative parsing context.

An MLE probability is the result of the positive counts divided by the candidate counts. For the PP-attachment model, positive counts are all cases from the training corpus that do attach, and candidate counts are the cases that do attach *plus* cases that could attach but that do not, according to the putative parsing context. For verb attachment (the relation label is $pobj$), then, candidate cases are all cases

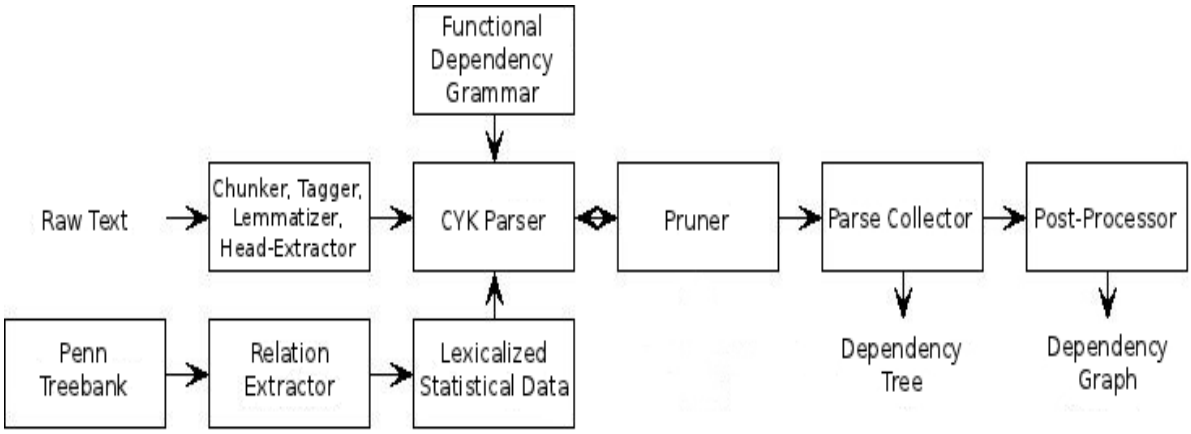


Figure 1: Pro3Gres flowchart

where attachment as $pobj$ occurs, *plus* all cases where in the ambiguous context of a verb-noun-PP sequence the PP attaches to the noun (the label is $modpp$).

$$p(pobj, dist|verb, prep, desc.noun) \cong \frac{\#(pobj,verb,prep,desc.noun)}{\#(pobj,verb,prep,desc.noun)+\#(modpp,verb,(\sum noun),prep,desc.noun)} \cdot \frac{\#(pobj,dist)}{\#pobj} \quad (3)$$

$$p(modpp, dist|noun, prep, desc.noun) \cong \frac{\#(modpp,noun,prep,desc.noun)}{\#(modpp,noun,prep,desc.noun)+\#(pobj,(\sum verb),noun,prep,desc.noun)} \cdot \frac{\#(modpp,dist)}{\#modpp} \quad (4)$$

(McDonald and Nivre, 2011) make a distinction between greedy, transition-based parsers like (Nivre, 2006) which take local decisions based on local state transitions (e.g. to shift or to reduce), and exhaustive graph-based parsers such as (McDonald et al., 2005) where (sub)graphs are modeled and many alternatives are kept. By their categorization Pro3Gres is an exhaustive graph-based parser. It uses a beam-search to discard unlikely partial analyses. Except for restrictions in the manually written grammar, the decisions of this parser are typically local. We will address this point in section 3.

The parser uses tagging and chunking as a preprocessing step, thus integrating fast finite-state techniques where appropriate, and converts dependency trees into graph structures in a post-processing step. The post-processing step includes the following incremental annotation: passive subjects are recognized, long-range dependencies are found, relative pronoun anaphora resolved, and verb-attached PPs are disambiguated between arguments and adjuncts.

An overview of the parser modules and their interactions is given in figure 1. We have chosen Pro3Gres for our experiments for the following reasons: (1) the strict separation into a manual grammar, which we have left unchanged, and a statistical disambiguation module is useful for our experiments, as it gives us control over the parameters, (2) as the parser uses explicit models and a restricted set of features it can be adapted fairly easily in order to conduct parsing experiments, (3) it shows a strong correlation between lexicalization and parsing quality, as we discuss in the following subsection.

1.2. The role of semantics for parsing

Bi-lexical statistics (Collins, 1999) has led to a quantum leap in parsing performance. But the debate on the importance of lexicalization is still open. On the one hand, decisions suffering from sparse data problems in the form of too little lexicalization lead to considerably worse results (e.g. (Collins and Brooks, 1995)), and approaches carefully extending lexicalisation can improve performance (McClosky et al., 2006; Stetina and Nagao, 1997). We have noticed a very strong correlation between backoff level and parser accuracy, as figure 2 illustrates for PP-attachment (nounpp = attachment of PP to a noun, verbpp=attachment of PP to a verb). Fully lexicalized decisions (Level 0: *head + preposition + description noun*), have much higher performance than those further down the back-off chain. Level 2 is *verb + preposition*, level 3 is *head class + preposition + noun*, level 4 is *verb class + preposition + description-noun class*, level 5 is *preposition + description-noun class*, level 6 is *preposition only*. We use the term description-noun to refer to the noun inside the PP.

On the other hand, (Gildea, 2001) have shown that monolexicalized approaches can perform almost as well. The approach of (Klein and Manning, 2003) is even unlexicalized; essentially it is an approach that uses semantic classes, stating that semantic classes can get one almost as far as pure bi-lexical preferences. One could tentatively summarize these opposing trends as follows: bi- and tri-lexicalized approaches can only perform well if data is not sparse, but data is sparse in the vast majority of cases. In those cases, a considerably less sparse good semantic classification can be as profitable. For this paper, it is tested in the following if there are semantics-based methods to reduce sparseness, so that more decisions can be taken at early backoff levels.

There are additional reasons why investigating the role of semantics for parsing is crucial. First, statistical approaches are now reaching a ceiling, although the error rate of even the best systems is still significantly and considerably higher than human inter-annotator disagreement. New sources of information need to be integrated. An obvious candidate for testing is semantics. Second, there are increasingly many approaches using syntactic modules for detection of thematic roles or doing syntactic parsing and thematic role detection simultaneously, see e.g. the CoNLL

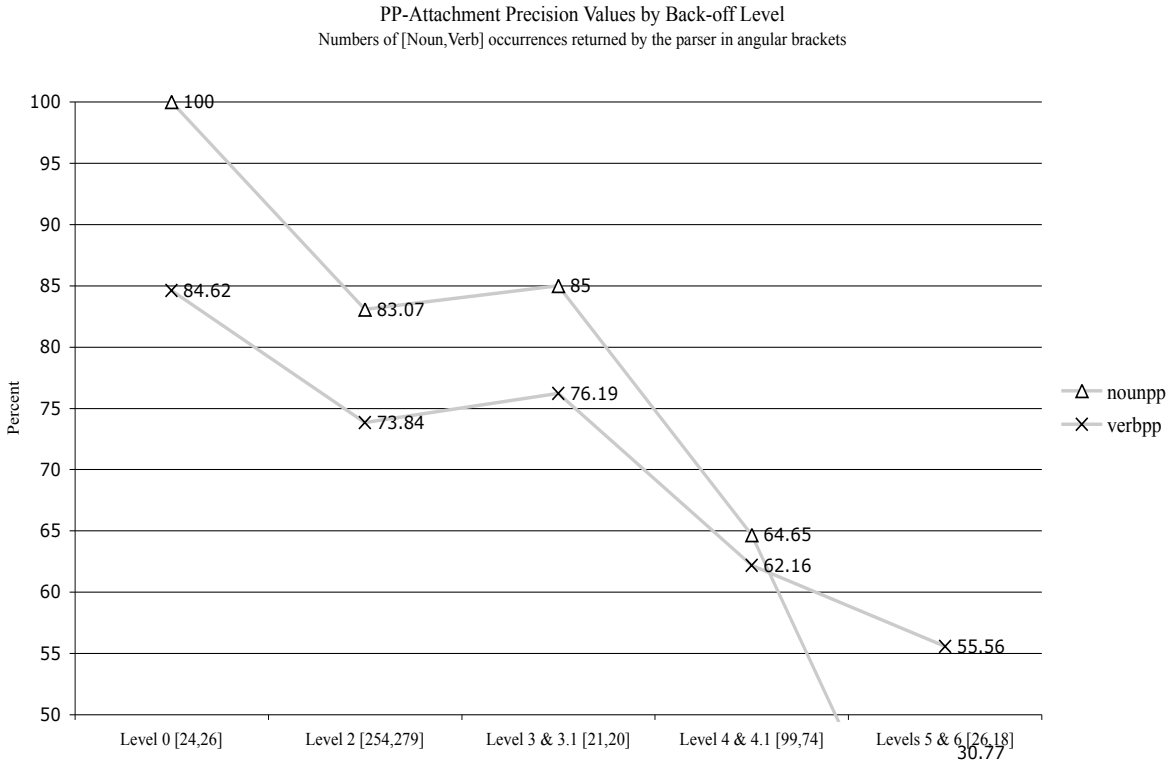


Figure 2: Evaluation: Quality of Backoff

2008 shared tass (Clark and Toutanova, 2008). Third, the types of errors that the various parsers make are often poorly understood. Investigating contributing factors, such as in (McClosky and Charniak, 2008) or even detailed error comparisons such as in (McDonald and Nivre, 2011) are very useful as they can help to disentangle lexical, syntactic and semantic factors.

In the rest of this paper, we will explore semantic factors to the end of increasing parsing performance. In section 2., we employ semantic information in the backoff system. In section 3., we use selectional restrictions and a non-local MLE model to boost plausible readings. We use semantic world-knowledge obtained from self-training in section 4. In section 5., we add an extension based on distributional similarity to the self-training model. Finally, we give an overview of the combined performance that we have gained due to our extensions in section 6.. We use GREVAL (Carroll et al., 2003) as evaluation corpus. It consists of 500 manually annotated sentences from the Susanne corpus.

2. Lexical semantic backoffs

We first report on experiments using semantic resources in the backoff.

2.1. Wordnet versus Levin class

We have discussed in the introduction that (Klein and Manning, 2003) have shown that a good semantic classification can get one as far as bi- and tri-lexicalized approaches. There are a number of semantic classification options for sparse data. We have used WordNet lexicographer file classes (Miller et al., 1990) as a simple approach, and alternatively Levin classes (Levin, 1993) for verbs. We compare the performance of these two ressources in figure 3. WordNet performs better in most cases. Also noun-PP attachment performance is indirectly affected. In order to break down performance across the whole confidence spectrum, we give threshold levels on the horizontal axis. The rightmost number, 0.9 means, for example, that only attachment decisions that were reported as being more than 90% probable in MLE attachment estimation (see introduction) were considered (which leads to high precision, but low recall). A potential reason why Levin classes perform worse is because their coverage is lower.

2.2. Similarity-based lexemes

We tested a number of extensions to fight the sparse data problem. In this section we employ an example-based use of the semantic constraints placed by syntactic relations. Because a head places strong selectional restrictions on its

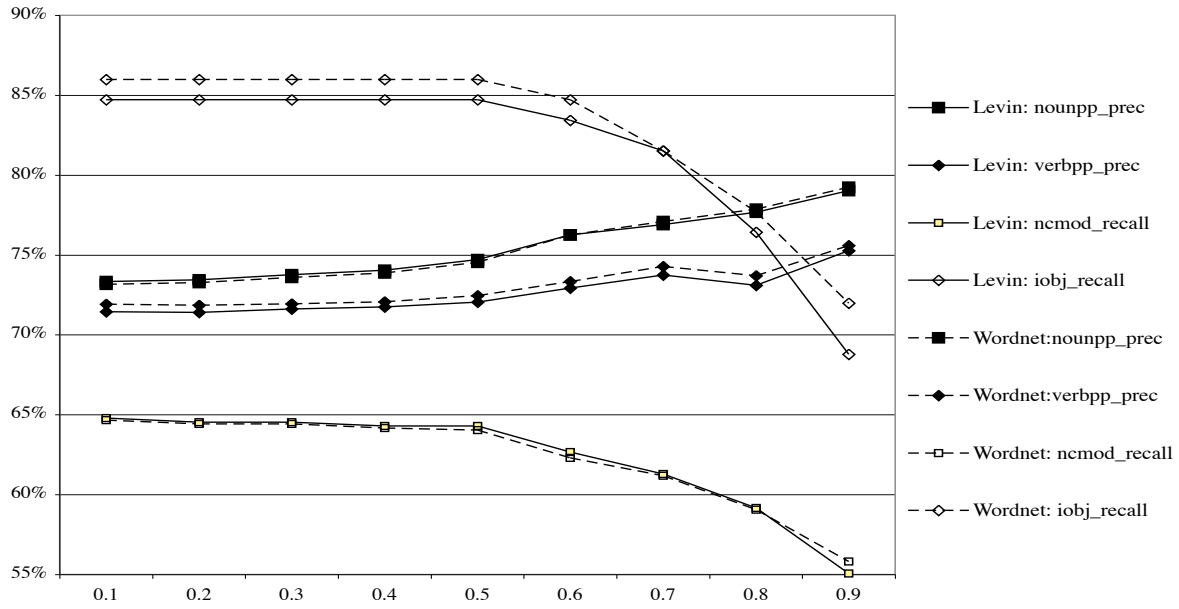


Figure 3: Comparison of Levin or Wordnet verb classes for backing off

dependent, dependents of the same head, or heads with the same dependent, are often similar. This fact can be exploited for Word Sense Disambiguation (e.g. (Lin, 1997)), the detection of similar words (Lin, 1998), WordNet synonyms (Curran, 2004) or distributional semantics vector models (Grefenstette et al., 2011). We use a very simple approximation here as follows:

For every target zero-count head-dependent pair, i.e. an attachment candidate at parse time for which we cannot find any occurrence at the first backoff level (the fully lexicalized level 0), if non-zero counts are found for both

1. a head'-dependent,
2. a head- dependent' and
3. a head'-dependent'

(where *head'* and *dependent'* are any word of the same tag as *head* and *dependent*, respectively), then their MLE counts are used. In a more restrictive version, only *dependent'* of the same WordNet noun class or verb class is allowed. Versions that use data from a large automatically parsed corpus (BNC) have also been tested. All of them show similar, slightly lower performance. An analysis of the decision points shows that non-zero values at between 2 and 10 times the original fully lexicalized level can be obtained, but the unreliability of the similarity and the increased coverage seem to level each other out. We assume that our first test was probably too simplistic. We will come back to this point again in section 5.

2.3. Unspecificity and probability

The level of backoff at which a decision can be taken is crucial as we have seen in figure 2. Better informed decisions are consistently better. At the first sight, informedness and probability seem unrelated. Informedness seems to have an impact on reliability and not on probability. On second thought, there is a reason why events are unseen – they are either indeed rare or simply impossible. The original parser only uses positive information. It also introduces artificial positive information in the form of smoothing, giving unseen events a low probability as is standardly done, but now we introduce positive information learning from the absence of word-word interactions.

From a probabilistic viewpoint, the negative information, although strictly speaking unquantifiable, that, whenever we can only decide late in the backoff chain, the fact that specific information is absent is an indirect indication that an event is indeed rare. In probability spaces where sparseness is relatively low, absence can be elevated to the status of partial evidence. If we had a complete system (closed world assumption), negative information (absence) could reliably be considered as positive information.

From a complementary distribution viewpoint, we have seen (figure 2) that there is a very strong relation between informedness expressed by the backoff level and performance. If a highly informed relation probability (say for verb PP-attachment) is in complementary distribution and hence competition with a less informed but equal probability (say for noun PP-attachment), we have evaluation performance statistics reasons to give preference to the highly informed relation.

PP-attachment	without "ironing"		with "ironing" (2%) = Base system	
subj_prec	849 of 946	89.75%	849 of 946	89.75%
local_subj_prec	826 of 912	90.57%	826 of 912	90.57%
subj_recall	855 of 1095	78.08%	855 of 1095	78.08%
obj_prec	353 of 412	85.68%	354 of 413	85.71%
obj_recall	351 of 428	82.01%	352 of 428	82.24%
nounpp_prec	351 of 497	70.62%	352 of 491	71.69%
verbpp_prec	353 of 477	74.00%	357 of 482	74.07%
ncmod_recall	530 of 801	66.17%	534 of 801	66.67%
iobj_recall	139 of 157	88.54%	140 of 157	89.17%
argmod_recall	34 of 40	85.0%	35 of 40	87.5%

Table 1: Results of evaluation with and without "ironing". Ironing takes unspecificity as expressed by backoff level as a punishing factor, we have used two 2% lower probability per backoff level

From a post-hoc performance perspective, there should be some way of taking the actual performance that is to be expected into consideration. With the benefit of hindsight, seeing that such an approach performs better, it makes sense to counter-balance obvious tendencies.
Although its status is probabilistically unclear, we have experimented with a simple extension for the PP-attachment relations that introduces an unspecificity punishment factor into the probability calculation. In our example, each probability is reduced by 2 percent for each backoff step. The results for some of the most frequent relations are given in table 1. Except for the subject relation, every relation shows an increase both in precision and in recall. The ambiguous PP-attachment relations profit in particular. The exact meaning of the labels is as follows:

- subj_prec , subj_recall: Precision and recall of the subject relation
- local_subj_prec: Precision of subject that are not in a long-distance relation, i.e. that are overtly expressed
- obj_prec , obj_recall: Precision and recall of the object relation
- nounpp_prec: Precision of the noun-PP attachment relation $modpp$
- verbpp_prec: Precision of the verb-PP attachment relation $pobj$
- ncmod_recall: Recall of PP adjuncts (mostly nominal, i.e. $modpp$)
- iobj_recall: Recall of PP arguments (mostly verbal, i.e. $pobj$)
- argmod_recall: Recall of *by*-agents in passive clauses (a part of $pobj$)

In distinction to smoothing, where positive information is produced, one could call this method *ironing*, because negative information irons out unwarranted and unjustified creases of too high probability caused by underspecificity. With values between 1 and 5%, "ironing" leads to better results, with values above that, results decline again. We use the model with 2% ironing as our base system for the following sections.
It has been shown for the fields of unsupervised grammar induction (Smith and Eisner, 2005) and for document classification (Schneider, 2004) that the ability of the classifier to use negative evidence makes a crucial difference in terms of performance.

3. Semantic Restrictions

In this section, we use selectional restrictions and a non-local MLE model to boost plausible readings.

3.1. Selectional Restrictions

We have discussed in section 1 that the original parser models probabilities using only those syntactic relations that are in competition. For example, every verb is in competition with one noun, the fact that several nouns may be in competition in a stacked NP is not modeled directly. Similarly, objects (e.g. *eat pizza*) and nominal adjuncts (e.g. *eat Friday*) are modeled as being in competition, but not subjects and objects. One could say that the original parser strictly models syntactic competition, to which we now add semantic competition. In the additionally introduced semantic probability model, every relation is in competition with every other relation. In order to calculate the probability for a verb-object relation between *rabbit* and *chase* we use the general probability of verb-object relation between *rabbit* and *chase* irrespective of which relations the object relation is in competition with. This has the effect that, in all likelihood, a sentence like *the rabbit chased the dog* gets a lower probability then *the dog chased the rabbit* because rabbits are very unlikely to be subjects of active instances of *chase*. Thus, our semantic world knowledge becomes part of the model, the parser parses for what is semantically more plausible. We will refer to this model as selectional restriction. While such an approach entails the risk of misinterpreting surprising new information, it is also psycholinguistically adequate: human parsers often disambiguate by using their expectations and their world knowledge. The results of the selectional restrictions model are given in table 2. The performance of almost every relation increases or stays unchanged.

3.2. Non-local Decisions

We have discussed in the introduction that the probabilities of the Pro3Gres parser are local, which means that world-knowledge expressed across more than one node generation is lost in the model. Although locality extends further in Dependency Grammar than in constituency grammar (where trees are more nested) and although there are global restrictions in the hand-written grammar, this is a serious shortcoming. In stacked PPs, for example, in the sequence verb-PP_1-PP_2 the attachment probabilities for verb-PP_1, verb-PP_2, and PP_1-PP_2 are only considered independently. It is well known that considering sister, grandmother and great-grandmother nodes increases parsing accuracy (e.g. (Charniak, 2000), (Bod et al., 2003)), particularly in the case of the highly ambiguous PP-attachment

Relation	without sel. rec. = Base system		with sel. rec.	
subj_prec	849 of 946	89.75%	854 of 950	89.89%
local_subj_prec	826 of 912	90.57%	830 of 916	90.61%
subj_recall	855 of 1095	78.08%	860 of 1095	78.54%
obj_prec	354 of 413	85.71%	354 of 414	85.51%
obj_recall	352 of 428	82.24%	352 of 428	82.24%
nounpp_prec	352 of 491	71.69%	353 of 486	72.63%
verbpp_prec	357 of 482	74.07%	358 of 480	74.58%
ncmod_recall	534 of 801	66.67%	535 of 801	66.79%
iobj_recall	140 of 157	89.17%	140 of 157	89.17%
argmod_recall	35 of 40	87.5%	35 of 40	87.5%

Table 2: Results of evaluation with and without selectional restrictions

PP-attachment	without multi-PP = Base system		with multi-PP	
nounpp_prec	352 of 491	71.69%	354 of 492	71.95%
verbpp_prec	357 of 482	74.07%	357 of 481	74.22%
ncmod_recall	534 of 801	66.67%	536 of 801	66.92%
iobj_recall	140 of 157	89.17%	140 of 157	89.17%
argmod_recall	35 of 40	87.5%	35 of 40	87.5%

Table 3: Results of evaluation with and without stacked PP model

relations. We have therefore added an MLE model which calculates the probabilities for verb-PP_1-PP_2 sequences and noun-PP_1-PP_2 sequences. For example, the probability that PP_2 is a dependent of PP_1 ($PP_1 < PP_2$) in a verb-PP-PP sequence, given the lexical items, is calculated as follows:

$$p(verb < (PP_1 < PP_2)) = \frac{\#(verb<(PP_1<PP_2))}{\#(verb<(PP_1<PP_2))+\#((verb<PP_1)<PP_2))}$$

The data is so sparse that in most cases only backoffs where all verbs and noun are replaced by their semantic verb- and noun-classes from Wordnet deliver results. The performance of the base system is compared to the new model in table 3, showing a slight improvement.

4. Distributional Semantics: Self-Training

The use of large amounts of parsed data is known as *self-training*. The variance of a large corpus is so big that it gives an opportunity to learn from the several different configurations, and parsing results from the many configurations with relatively low ambiguity may deliver a signal that is strong enough. In a nutshell, self-training can improve results where sparseness is worse than error rate. From a semantic viewpoint, parsing large amounts of data allows us to learn world knowledge from the distribution of syntactic relations. The main danger of self-learning is that the ensuing corpus skew will lead to the same problems as in co-training (Sarkar, 2001) and boost errors. Until recently, self-training was thought to be unable to lead to better performance (Charniak, 1997; Steedman et al., 2003). (Bacchiani et al., 2006) have shown that self-training can improve parsing out-of-domain texts, and is therefore a suitable approach for domain adaptation. (McClosky et al., 2006) was the first approach to show that the use of a re-ranker (Charniak and Johnson, 2005) can also improve in-domain parsing. Their re-ranker uses a very rich set of features, which leads to a sufficiently different view on the data to allow for an increase in performance.

(McClosky et al., 2008) describe some of the reasons that lead to an improvement from self-training. They reject the assumptions that high performance of the underlying parser is a prerequisite and that analyses that are missed by the underlying parser are a problem. They find out that two major sources of improved performance are (1) the different view on the data and (2) the reduction of sparseness: bi-lexical heads that are unseen in the Penn Treebank but seen in the self-training lead to a clear improvement: *"H (biheads) is the strongest single feature and the only one to be significantly better than the baseline"* (p. 567). This indicates that the debate on the importance lexicalization is still open.

A reliable measure of confidence on whether a parser decision is correct or not plays a crucial role in self-training. If this measure were completely reliable, only correct parses would be added to the training corpus. The parser which we use offers a sufficiently good measure: there is a very strong correlation between backoff level and the correctness of the parser decision, as figure 2 shows. This can be exploited, e.g. by adding self-training results late in the backoff chain, thus using tri- or bi-lexical self-training decisions if the Penn Treebank training data only offers monolexical decisions.

The Penn Treebank contains 1 million words. We have parsed the 100 million words British National Corpus BNC (Aston and Burnard, 1998), which gives us 2 orders of magnitude more lexicalized data to alleviate the sparse data problem. The PP-attachment error rate on the BNC is clearly lower than the error rate on PP-attachment cases from low backoff-levels (figure 2). We have added the self-trained counts into the backoff hierarchy between level 2 and 3. The results are given in table 4. There is a small increase in the PP-attachment relations. The increase is too small to be statistically significant, however, so it can only serve as an indication. Therefore, a larger evaluation corpus will be needed. There are only 43 cases in GREVAL in which the top-ranked reading includes a decision from the new self-trained backoff level, which means that we obtain 3 improvements out of 43 cases.

Most approaches to self-training use a re-ranker, e.g. (McClosky et al., 2006) as a crucial element. We have presented an approach which does not need a re-ranker but improves performance. It is known that co-training (Sarkar, 2001; Hwa et al., 2003) only leads to minimal improvements. Our approach is different from co-training for a number of reasons: (1) for highly informed levels, we only use the original training set, and (2) we retain all parses, which reduces the risk of skewing the corpus or disappearing into an "error hole" as it can typically happen in co-training.

Relation	without BNC self = Base system		with BNC self	
subj_prec	849 of 946	89.75%	849 of 946	89.75%
local_subj_prec	826 of 912	90.57%	826 of 912	90.57%
subj_recall	855 of 1095	78.08%	855 of 1095	78.08%
obj_prec	354 of 413	85.71%	354 of 413	85.71%
obj_recall	352 of 428	82.24%	352 of 428	82.24%
nounpp_prec	352 of 491	71.69%	353 of 492	71.75%
verbpp_prec	357 of 482	74.07%	357 of 481	74.22%
ncmod_recall	534 of 801	66.67%	534 of 801	66.67%
iobj_recall	140 of 157	89.17%	140 of 157	89.17%
argmod_recall	35 of 40	87.5%	36 of 40	90.0%

Table 4: Results of evaluation with and without self-training

5. Combining self-training and example-based similarity

We have learnt in the previous section that self-learning can work if we have a reasonably reliable measure indicating where sparse data leads to errors. Such a measure can be obtained from the backoff level, and thus we use self-training decisions only for late backoff instances. We have learnt in section 2 that simplistic "naive" approaches to distributional similarity do not work. We have used similarity-based counts directly after the the fully lexicalized level 0. The imprecision that such a simplistic similarity approach introduces is probably still higher than the error rate at the second-highest backoff level. We thus re-delegate the similarity-based approach to the level after the BNC-self-trained data. The data from the parsed BNC is used, and the restrictive version, in which only *head'* and *dependent'* of the same WordNet noun class or verb class as *head* and *dependent*, respectively, is allowed. Performance is very similar to the self-trained model in the previous section.

We have made a further restrictions: similarity-pairs (*head'-dependent, head'-dependent'* and *head'-dependent'*) are generated from the BNC, but only MLE probabilities from the error-free Penn Treebank are allowed, i.e. if the Penn treebank contains data for a *head'-dependent* or *head-dependent'* pair it is taken, otherwise the backoff chain continues resorting to the next, lower level. Results are given in table 5, comparing the self-trained model to the self-trained similarity model. We have added this extension only to the PP-attachment relations. Again, the improvement is probably strictly speaking not statistically significant. In the GREVAL corpus, there are 7 cases that improve. There are only 13 cases, however, in which the top-ranked reading includes a decision from the new self-trained plus similarity backoff level, which means an improvement of 7 out of 13.

We would like to use a vector-based semantics model in future research, for example (Grefenstette et al., 2011). The current pilot study has shown that a gain in parsing performance from using similarity-based metrics against sparse data can be expected.

Relation	BNC self =right col. of table 4		BNC self + similarity	
nounpp_prec	353 of 492	71.75%	356 of 494	72.06%
verbpp_prec	357 of 481	74.22%	357 of 479	74.53%
ncmod_recall	534 of 801	66.67%	538 of 801	67.17%
iobj_recall	140 of 157	89.17%	140 of 157	89.17%
argmod_recall	36 of 40	90.0%	36 of 40	90.0%

Table 5: Results of evaluation with original self-training and with added example-based similarity

Relation	Base System		Combined	
subj_prec	849 of 946	89.75%	854 of 950	89.89%
local_subj_prec	826 of 912	90.57%	830 of 916	90.61%
subj_recall	855 of 1095	78.08%	860 of 1095	78.54%
obj_prec	354 of 413	85.71%	354 of 414	85.50%
obj_recall	352 of 428	82.24%	352 of 428	82.24%
nounpp_prec	352 of 491	71.69%	359 of 491	73.12%
verbpp_prec	357 of 482	74.07%	357 of 475	75.16%
ncmod_recall	534 of 801	66.67%	541 of 801	67.54%
iobj_recall	140 of 157	89.17%	140 of 157	89.17%
argmod_recall	35 of 40	87.5%	35 of 40	87.5%

Table 6: Evaluation comparison between base system and combined additions

6. Combined Model and Discussion

Finally, we give an overview of the combined performance that we have gained from the extensions introduced in sections 3 to 5. The results are given in table 6. Performance remains unchanged in 3 lines, there is one slight decline (*obj* recall), PP-attachment precision increases by over a percent, while recall also slightly improves. In terms of parsing speed, the extensions made in sections 4 and 5 are costly. The original parser parses the 500 sentence GREVAL corpus in under a minute, and the 100 million words BNC in about a day. Parsing times in sections 2 and 3 hardly change, in section 4 it increases to about a minute and to about 5 minutes in section 5.

While a performance increase of maximally 1.5% may seem very moderate, it should be considered in view of the law of diminishing marginal utility, in comparison to the baseline and the upper bound, and supplemented with an analysis of errors. For this discussion, we will focus on the PP-attachment relations.

As a PP-attachment baseline model, we use a version of the parser that uses the base system for all relations, but for the PP-attachment relations it only uses the preposition, i.e. backoff level 6. Results are given in table 7, first column (*Baseline*). In terms of precision, the increase from the base system to the combined system is as big as the one from baseline to base system, about 1.4%. In terms of recall, the increase from the baseline to the base system is 2.4%, the increase from the base system to the combined system is another 0.7%.

As PP upper bound, we use version of the combined system that reports not only the top ranked, but the first 64 readings for every sentence. While precision is negatively affected by a random element, the recall thus obtained gives one an

Relation	Baseline		Base System		Combined		Upper Bound	
nounpp_prec	337 of 472	71.40%	352 of 491	71.69%	359 of 491	73.12%	–	–
verbpp_prec	358 of 501	71.46%	357 of 482	74.07%	357 of 475	75.16%	–	–
ncmod_recall	517 of 801	64.54%	534 of 801	66.67%	541 of 801	67.54%	630 of 801	78.65%
iobj_recall	139 of 157	88.54%	140 of 157	89.17%	140 of 157	89.17%	144 of 157	91.71%
argmod_recall	39 of 40	97.50%	35 of 40	87.50%	35 of 40	87.50%	40 of 40	100%
$\sum$ PP Prec	695 of 973	71.43%	709 of 973	72.87%	716 of 966	74.12%	–	–
$\sum$ PP Recall	695 of 998	69.64%	709 of 998	71.04%	716 of 998	71.74%	814 of 998	81.56%

Table 7: Evaluation comparison for PP-attachment relations between baseline, base system, combined additions and upper bound

Relation	Attachment Error	Head Extraction Error	Chunking or Tagging	compl/prep Error	Grammar Mistake or incompl. Parse	Grammar Assumption
Noun-PP Precision	22	1	8	0	3	3
Noun-PP Recall	25	1	14	0	12	5
Verb-PP Precision	12	1	5	1	1	2
Verb-PP Recall	2	0	1	0	0	0
Totals	61	3	28	1	16	10
Proportions	51 %	3 %	24 %	1 %	13 %	8 %

Table 8: Detailed Analysis of the PP-attachment errors in the first 100 evaluation corpus sentences

assessment of the how accurate results can get if an oracle ranked all possible readings correctly. The recall measures are given in table 7, last column (*Upper Bound*), showing that the 1% improvement in $ncmod\_recall$ corresponds to almost a tenth of the maximally possible increase.

An analysis of PP-attachment errors in table 8 shows why almost a fifth of $ncmod$ cannot be found. We have investigated the PP-attachment errors in the first 100 sentences in the 500 sentence evaluation corpus (GREVAL, (Carroll et al., 2003)) in (Schneider, 2008), according to the output of the base system. About half of the errors are attachment errors, almost a quarter are chunking or tagging errors. Grammar mistakes or incomplete parses are cases which the grammar did not handle correctly, for example because the grammar does not allow X-bar violations and places strong restrictions PPs that precede their governor. The category of grammar assumption involves cases where our intended analysis as mirrored in our grammar does not coincide with the grammar view of the gold standard annotators. The majority of attachment errors can be corrected by selecting the correct non-first analysis, other errors cannot be corrected by our current parser.

7. Conclusion

We have successfully used several semantic resources to improve the performance of a syntactic dependency parser and have learnt a number of things on the way. We have learnt in section 2 that our first very simple approach to using similarity-based measures does not improve performance. We have learnt that Levin classes lead to a smaller improvement than WordNet classes. We have seen that negative information can up to a point be used as partial evidence. Although its probabilistic status is unclear, punishing late backoff decisions considerably improves performance. We have called our approach *ironing* because negative information irons out unwarranted and unjustified creases of too high probability caused by underspecificity.

In section 3, we have employed selectional restrictions to boost interpretations that are semantically plausible. We have also added an MLE model considering grandmother and sister node information for PP attachment in order to be able to profit form world knowledge that is expressed across two node generations. Both extensions increase performance.

In section 4, we have presented an approach using self-training which does not need a re-ranker, unlike e.g. (McClosky et al., 2006), and shown that it leads to improved performance. We use a parser which delivers a relatively reliable measure of parsing quality (figure 2), which we can exploit. We have learnt that self-training can work if we apply it only in those cases where we know that the expected backoff performance is lower than general parser performance.

In section 5, we use what we have learnt in section 4 to improve our simple distributional semantics approach to detect similar words. If we constrain our criteria to detect similar words, use only MLE counts from the Penn Treebank, and add the model late in the backoff chain (where decisions are of relatively poor quality) we gain a considerable improvement in parsing quality.

Finally, we combine the improvements made in sections 3 to 5. Particularly the ambiguous PP-attachment relations improve. PP-attachment precision improves by over 1% while also recall improves slightly. We discuss the performance in comparison to a baseline and the upper bound and give a brief error analysis.

An additional conclusion that we can draw from the current pilot study is that employing semantic resources has the potential to increase the performance of parsers considerably. More systematic approaches, for example using

vector-space models (Grefenstette et al., 2011) and large evaluation corpora will be used in future research.

8. References

Guy Aston and Lou Burnard. 1998. *The BNC Handbook. Exploring the British National Corpus with SARA*. Edinburgh University Press, Edinburgh.

Michiel Bacchiani, Michael Riley, Brian Roark, and Richard Sproat. 2006. MAP adaptation of stochastic grammars. *Computer Speech and Language*, 20(1):41–68.

Rens Bod, Remko Scha, and Khalil Sima'an, editors. 2003. *Data-Oriented Parsing*. Center for the Study of Language and Information, Studies in Computational Linguistics (CSLI-SCL). Chicago University Press.

John Carroll, Guido Minnen, and Edward Briscoe. 2003. Parser evaluation: using a grammatical relation annotation scheme. In Anne Abeillé, editor, *Treebanks: Building and Using Parsed Corpora*, pages 299–316. Kluwer, Dordrecht.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 173–180, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proc. of the 15th Annual Conference on Artificial Intelligence (AAAI-97)*, page 598603, Stanford, USA.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the North American Chapter of the ACL*, pages 132–139.

Alexander Clark and Kristina Toutanova, editors. 2008. *CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning*. Coling 2008 Organizing Committee, Manchester, England, August.

Michael Collins and James Brooks. 1995. Prepositional attachment through a backed-off model. In *Proceedings of the Third Workshop on Very Large Corpora*, Cambridge, MA.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA.

James R. Curran. 2004. *From Distributional to Semantic Similarity*. Doctoral thesis, Institute for Communicating and Collaborative Systems, University of Edinburgh.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 167–202, Pittsburgh, PA.

Edward Grefenstette, Mehrnoosh Sadrzadeh, Stephen Clark, Bob Coecke, and Stephen Pulman. 2011. Concrete sentence spaces for compositional distributional models of meaning. In *Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)*, Oxford.

Katri Haverinen, Filip Ginter, Sampo Pyysalo, and Tapio Salakoski. 2008. Accurate conversion of dependency parses: targeting the Stanford scheme. In Tapio Salakoski, Dietrich Rebholz-Schuhmann, and Sampo Pyysalo, editors, *Proceedings of Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008)*, pages 133–136, Turku, Finland. Turku Centre for Computer Science (TUCS).

Susan Hunston and Gill Francis. 2000. *Pattern Grammar: A Corpus-Driven Approach to the Lexical Grammar of English*. Benjamins, Amsterdam/Philadelphia.

Rebecca Hwa, Miles Osborne, Anoop Sarkar, and Mark Steedman. 2003. Corrected co-training for statistical parsers. In *Proceedings of the ICML Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining at the 20th International Conference on Machine Learning (ICML-2003)*, Washington DC.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan.

Beth C. Levin. 1993. *English Verb Classes and Alternations: a Preliminary Investigation*. University of Chicago Press, Chicago, IL.

Dekang Lin. 1997. Using syntactic dependency as local context to resolve word sense ambiguity. In *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 64–71, Madrid. Association for Computational Linguistics.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association of Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL '98)*, pages 768–774, Montreal.

Mitch Marcus, Beatrice Santorini, and M.A. Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19:313–330.

David McClosky and Eugene Charniak. 2008. Self-training for biomedical parsing. In *Proceedings of ACL-08: HLT, Short Papers*, pages 101–104, Columbus, Ohio, June. Association for Computational Linguistics.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Reranking and self-training for parser adaptation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 337–344, Sydney, Australia, July. Association for Computational Linguistics.

David McClosky, Eugene Charniak, and Mark Johnson. 2008. When is self-training effective for parsing? In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 561–568, Manchester, UK, August. Coling 2008 Organizing Committee.

Ryan McDonald and Joakim Nivre. 2011. Analyzing and integrating dependency parsers. *Computational Linguis-*

tics, 37(1):197–228.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 523–530, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Paola Merlo and Eva Esteve Ferrer. 2006. The notion of argument in PP attachment. *Computational Linguistics*, 32(2):341 – 378.

George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3:235–244.

Joakim Nivre. 2006. *Inductive Dependency Parsing*. Text, Speech and Language Technology 34. Springer, Dordrecht, The Netherlands.

Anoop Sarkar. 2001. Applying co-training methods to statistical parsing. In *Proceedings of NAACL 2001*, Pittsburgh, PA.

Karl-Michael Schneider. 2004. On word frequency information and negative evidence in naive bayes text classification. In *Proceedings of España for natural language processing, ESTAL*.

Gerold Schneider. 2008. *Hybrid Long-Distance Functional Dependency Parsing*. Doctoral Thesis, Institute of Computational Linguistics, University of Zurich.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–124.

John Sinclair. 1991. *Corpus, Concordance, Collocation*. OUP, Oxford.

Noah A. Smith and Jason Eisner. 2005. Guiding unsupervised grammar induction using contrastive estimation. In *International Joint Conference on Artificial Intelligence (IJCAI) Workshop on Grammatical Inference Applications*, pages 73–82, Edinburgh, July.

Mark Steedman, Steven Baker, Jeremiah Crim, Stephen Clark, Julia Hockenmaier, Rebecca Hwa, Miles Osborne, Paul Ruhlen, and Anoop Sarkar. 2003. Semi-supervised training for statistical parsing. Technical Report CLSP WS-02 Final report, John Hopkins University.

Jiri Stetina and Makoto Nagao. 1997. Corpus based PP attachment ambiguity resolution with a semantic dictionary. In *Proceedings of the Fifth Workshop on Very Large Corpora*, pages 66–80, Beijing and Hong Kong, Aug. 18 – 20.

Using explicitly and implicitly encoded semantic relations to map Slovene Wordnet and Slovene Lexical Database

Darja Fišer[1], **Polona Gantar**[2], **Simon Krek**[3]
[1]Department of Translation, Faculty of Arts, University of Ljubljana
Aškerčeva 2, 1000 Ljubljana, Slovenia
[2]Scientific Research Centre of the Slovenian Academy of Sciences and Arts Ljubljana
Novi trg 4, 1000 Ljubljana, Slovenia
[3]"Jožef Stefan" Institute, Artificial Intelligence Laboratory
Jamova cesta 39, SI-1000 Ljubljana, Slovenia
[1]darja.fiser@ff.uni-lj.si, [2]apolonija.gantar@guest.arnes.si, [3]simon.krek@guest.arnes.si

Abstract

In this paper we present the results of a case study in which we use explicitly and implicitly encoded semantic relations to automatically map lexical entries from two different lexical semantic resources for Slovene with the well-known Simplified Lesk algorithm. We explain the selection of the mapping sample and mapping elements and describe the pre-processing steps that were performed in order to facilitate the mapping procedure. Manual evaluation of the mappings shows promising results, especially for nouns which were correctly mapped in 68% of the cases. Discrepancies in the mappings are also analysed in order to gain insight into the conceptual differences between the resources and investigate possible future refinements of the mapping procedure.

Keywords: lexical semantics, semantic relations, automatic mapping of lexical resources, polysemy, wordnet

1. Introduction

A wide range of lexical resources have been created to support natural language processing and it is commonly accepted that we could benefit from merging the lexical information each one of them contains into an even larger and richer knowledge base. However, since each resource has been created for a different purpose, they also have practical and theoretical peculiarities that make it difficult to combine the information from the different resources (Loper et al., 2007).

Wordnet is an extremely popular lexico-semantic resource and as such it is unsurprising that it has been automatically mapped to many other resources. Most well-known formal ontologies have been mapped to wordnet, such as SUMO (Niles and Pease, 2003), DOLCE (Gangemi et al., 2003), Cyc (Reed and Lenat, 2002) and UMLS (Burgun and Bodenreider, 2001). More recently, wordnet has also been merged with the collaboratively created Wikipedia (Suchanek et al., 2008) as well as to FrameNet (Tonelli and Pianta, 2009) and other verb databases (Green et al., 2001).

While it is true that, unlike for English, there are still very few lexical resources available for Slovene that would call for such attempts, two highly valuable, potentially complementary, resources have recently been developed: a corpus-based lexical database of Slovene, the primary goal of which is to serve as a foundation of a new generation of Slovene dictionaries, and an automatically created wordnet for Slovene, the aim of which is to enhance semantic processing of Slovene texts in various tasks, such as automatic word-sense disambiguation, information retrieval and machine translation. While they are both based on the word sense principle, they focus on different types of lexical information and could therefore be mutually beneficial if merged into a single resource.

The Slovene Lexical Database, on the one hand, provides very useful collocations, usage examples and lexico-syntactic patterns that would be a welcome addition to Slovene wordnet which, on the other hand, has plenty to offer in terms of semantic and lexical relations that are missing in the lexical database as well as equivalence links to translations of the same concepts in other languages that would upcycle an essentially monolingual resource into a bi- or multilingual one at a relatively low cost.

This is why the primary aim of this paper is to examine in what way and to what extent these two resources could be merged in an automated way on a sample of lexemes which are present in both resources. The mapping procedure will be supported by semantic relations, which are encoded explicitly in Slovene wordnet and implicitly throughout the structure of the lexical entry in the Slovene Lexical Database.

Since the merging attempts will no doubt reveal incongruences, our secondary goal is to analyze them in order to identify and improve weaknesses of one or the other resource, both of which are still under development. With the analysis of the mappings we wish to fine-tune future development of the two resources as well as establish a large-scale mapping procedure that would encompass entire databases.

The rest of this paper is organized as follows: in the next two sections we present the Slovene Lexical Database and the Slovene Wordnet, in Section 4 we describe the mapping procedure, in Section 5 we analyze and discuss the results after which we wrap up the paper with some concluding remarks and plans for future work.

2. Slovene Lexical Database

The Slovene lexical database (SLD) is a lexical resource with dictionary-type of information on words and word combinations (senses, collocations, examples, syntactic patterns, grammatical information etc.). It is being compiled within the "Communication in Slovene"

project (Gantar and Krek, 2011) from 2008-2012. It will also be used to enhance natural language processing tools for Slovene. Information from the SLD together with the complementary morphological lexicon data and other resources will be integrated in an interactive web portal intended for pupils and students as well as for general users. The database was compiled from the Gigafida corpus (Logar and Krek, 2010), a new generation of Slovene corpora which contains 1.1 billion words from texts of different genres, including Internet content, spanning from 1990-2010.

SLD contains two types of information which are intended for two types of users: the first is the lexico-grammatical information that is intended for human users and comes in the form of sense descriptions which broadly follow the principles of Frame semantics and represent the starting point for whole-sentence definitions. Also included are collocations and typical examples from the corpus, which are both attributed to particular senses and syntactic patterns of the lemma. The second type of information are designed for natural language processing tools. Among them are the formal encoding of syntactic patterns at the clause and phrasal level (syntactic structures) as well as the formal encoding of semantic arguments and their types.

The database is conceptualized as a network of interrelated lexico-grammatical information on six hierarchical levels with the semantic level functioning as the organizing level for the subordinate syntactic and collocation levels.

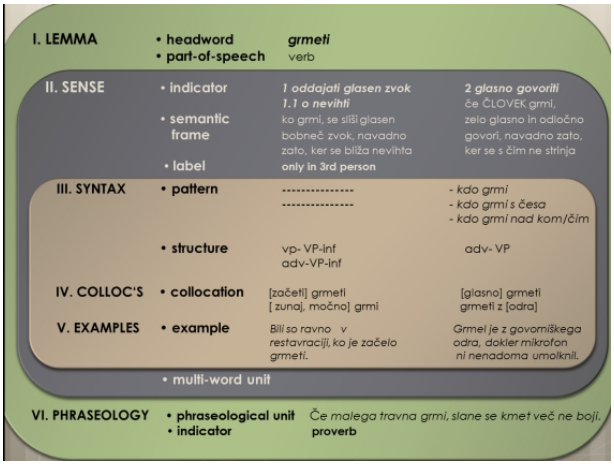


Figure 1: Six levels of description in SLD

On the first level, senses and subsenses of the lemma are specified. All senses and subsenses are labelled with semantic indicators (in the form of simple EFL dictionary-like explanations or synonyms) whose primary function is to form a sense menu intended for easy navigation within a polysemous entry structure. Each sense or subsense can be qualified with a domain, register, style or similar label. Another kind of information recorded on the sense level are semantic frames similar to FrameNet (Fillmore et al., 1992; Baker et al., 2003) and the prototypical syntagmatic patterns in the Corpus Pattern Analysis system (Hanks 2004). The semantic frames are used to record argument structure and semantic types found in a particular sense or subsense in a form of if-clauses similar to whole-sentence definitions in COBUILD dictionaries (Barnbrook, 2002) which include information about typical syntactic patterns, reflexivity, pragmatic aspects of headword usage, or grammatical limitations. Semantic types are linked to other kinds of information on subordinate levels. On the collocation level, for example, patterns and structures are verified through corpus data by recording typical collocates of the headword realized in the anticipated syntactic positions.

Multi-word expressions are included either within a particular sense/subsense or below all the senses and subsenses, and are described by a semantic indicator, mostly identifying a broad semantic field or domain.

SLD data is being collected from the corpus with the Sketch Engine system (Kilgarriff and Tugwell, 2001), a popular lexicographic corpus data extraction tool that enables faster compilation of the database. Apart from the standard and advanced use of the concordancer, two additional features are used. The first one is the word sketches module that is based on the Slovene sketch grammar with 32 grammatical relations (Kilgarriff and Krek, 2006) which reflect the 300 recorded syntactic structures. The other feature are the combined Tickbox lexicography and GDEX modules which provide a faster way to select good dictionary examples. The module has been adapted for Slovene (Kosem et al., 2011).

In Table 1 some figures from the current version of SLD are presented. They show that SLD currently contains 2,300 entries that are split into 1.74 senses on average or 3.02 combined senses and subsenses per entry. There are 43,618 collocations and 11,994 multi-word expressions in total, or 18.9 collocations and 5.20 multi-word expressions per entry. The number of corpus examples is quite high: 55.13/entry or 2.92/collocation.

entries	2,308
senses	4,012
subsenses	2,952
collocations	43,618
examples	127,239
multi-word expressions	11,994
labels	962
phraseological units	2,120

Table 1: Some figures from SLD

Semantic relations are not explicitly included in the current version of SLD, although it is planned that such relations will be established post-festum through a consolidation of data on the level of semantic indicators present in each sense, subsense, multi-word expression and phraseological unit. Mapping with wordnet synsets is one method by which such consolidation can be achieved, in addition to making two extensive databases compatible from which both their human users and computer applications will profit.

3. Slovene Wordnet

Slovene Wordnet (sloWNet) is a semantic lexicon that is based on the Princeton WordNet for the English language (Fellbaum, 1998). In it, nouns, verbs, adjectives and adverbs are grouped into sets of synonyms (e.g. *{car, automobile}*) which are then organized into a hierarchical network with lexical and semantic relations, such as hyper- and hyponymy, antonymy, meronymy etc. (e.g. *{car, automobile} HYPERNYM-> {vehicle}*). mThe concepts that synonym sets (synsets) represent are defined with a short gloss and usage examples while most synsets also have a domain label and a mapping to the SUMO/MILO ontology.

sloWNet was constructed automatically by leveraging existing bi- and multilingual resources, such as a bilingual dictionary, a multilingual parallel corpus and encyclopaedic resources from the Wikipedia family. Based on the assumption that the translation relation is a plausible source of semantics (Dyvik, 1998) and that it will reveal words which can have more than one meaning on the one hand and different expressions that share the same meaning on the other, we have used these resources in combination with BalkaNet wordnets (Tufis et al., 2000) to extract semantically relevant information in three different approaches we briefly describe below.

Slovene wordnet was built automatically in three stages, each using a different approach according to the resources used for extracting the relevant lexico-semantic information. The first and most straightforward approach relied on the Serbian wordnet (Krstev et al., 2004) where the literals were translated into Slovene utilizing a traditional digitized bilingual Slovene-Serbian dictionary (Erjavec and Fišer, 2006). This simple approach lacked automatic disambiguation of polysemous dictionary entries and therefore required a lot of manual cleaning. This was improved in the second approach which was able to assign the correct wordnet sense to a Slovene equivalent by disambiguating it with a word-aligned parallel multilingual corpus and already existing wordnets for several languages (Fišer, 2007). The main contribution of the third approach was the extraction of a large number of monosemous specialized vocabulary and multi-word expressions from Wikipedia and its related resources (Fišer and Sagot, 2008).

The next major step in the development of sloWNet 3.0 is the recent large-scale automatic extension in which we combined all the resources from the previous steps in order to exploit the available resources to their full potential and thereby improve coverage of sloWNet without compromising its quality. First, a model was trained on the existing elements in sloWNet, and a maximum entropy classifier was used to determine appropriate senses of translation candidates extracted from the heterogeneous resources described above (see Sagot and Fišer, 2012).

The extended sloWNet has 82,721 literals, which are organized into 42,919 synsets. Apart from single words sloWNet contains many multi-word expressions and proper names as well. Nouns are still by far the most frequent, representing more than 70% of all synsets. While 66% of all the literals in sloWNet are monosemous, their average polysemy level is 2.07.

sloWNet can be viewed in sloWTool we designed for browsing, editing and visualizing wordnet content (Fišer and Novak, 2011). An example of a Slovene synset with its corresponding English equivalent as displayed in sloWTool can be seen in Figure 1. In addition to Slovene and English synonyms describing a concept, a definition is given, after which relations pointing to semantically related synsets are shown.

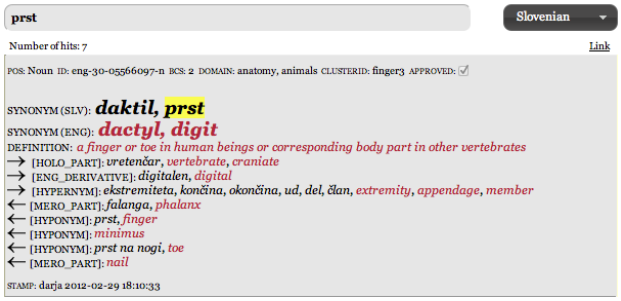


Figure 2: A sloWNet synset in sloWTool.

The set of approaches we used to create sloWNet have two important consequences for our mapping task: first, since the lexicon was created automatically, the generated synsets contain some noise which could have a negative impact on the mapping process. In order to minimize it, we have manually examined and corrected any mistakes in all the synsets containing the words we focus on in this experiment. And the second consequence, which will leave a much more permanent mark on the merged resource, is that the organization of the senses and the semantic network is English-centered and might therefore include some irrelevant concepts for Slovene or miss others that play an important role in Slovene language and culture, making the mapping difficult.

4. Experimental setup

4.1 Mapping sample

In this pilot study we performed the mapping on a sample of lexical entries, which were carefully selected in order to resemble the large-scale mapping of the entire databases in the future as closely as possible. Since sloWNet currently contains very few adverbs, the sample comprises 10 general-language, single-word nouns, adjectives and verbs respectively, which existed in both recourses and were also polysemous in both of them.

Although SLD organizes lexical entries into a hierarchy of senses and subsenses, we treat them all as individual senses in this experiment and try to find the best fit among the wordnet senses containing the same literal for each of them, not only for the main senses. Taking this into account, the lowest polysemy level of

the words included in the sample was 3 senses per word in SLD and 2 senses per word in sloWNet (*icon*). The highest polysemy level was 12 senses in SLD and 15 in sloWNet (*play*). Overall, the average polysemy level was 6.5 in SLD and 5.5 in sloWNet. In SLD, the average level of polysemy was much higher for verbs (7.5) while it was more or less the same for all parts of speech in sloWNet.

Polysemy level	SLD senses	SWN senses
2-3 senses	2	10
4-5 senses	12	6
6-10 senses	10	11
11-14 senses	6	3
Total	30	30

Table 2: Level of polysemy for sample words in both resources

However, the mere number of senses often does not give the complete picture of the difficulty of a task in lexical semantics, since it is the kind of sense distinctions that really matter in many cases. This is why we made sure to include words displaying coarse- as well as fine-grained polysemy (e.g. homonymous word *prst* which can mean *finger* or *soil* vs. polysemous word *jagoda* which can mean either *strawberry* the plant or the fruit it bears), expecting that the coarsely-grained senses will have a higher mapping accuracy than the finely-grained ones. In addition, we tried to include words denoting concrete as well as abstract concepts in order to be able to analyse the impact of concept-defining features on the quality of the mapping.

4.2 Mapping elements

After having selected the sample of words to be mapped between the two resources, we examined the structure of the lexical entries in both resources in order to determine the elements that are on the one hand the most indicative for the intended sense in a given resource and most comparable across the resources on the other.

This analysis showed that it is the lexical and semantic relations that are shared to the highest degree in both resources with an important distinction that they are in most cases explicitly encoded in sloWNet but only implicitly used in several elements in SLD. For example, in sloWNet, the synset for *horse* points to a more general concept *animal* directly via the hypernymy relation while in SLD the same more general concept is referred to in the element called <indicator> containing a short gloss for the word sense it describes.

In a similar fashion, overlapping semantically related words are frequently found in FrameNet-like semantic descriptions given in the element <semantic_frame> and/or the <definition> element which often contain hypernyms, co-hyponyms as well as meronyms but also in some other elements, such as <collocation> and <multiword_combination> which often contain hyponyms, and in the <label> element which gives the domain the word sense is usually used in.

SLD elements	Contain relations	SWN elements	Contain relations
<indicator>	*hypernym domain*	<synonym>	synonym
<semantic_ frame>	*hypernym cohyponym meronym holonym derivation*	<definition>	*hypernym cohyponym meronym holonym derivation*
<definition>	*hypernym cohyponym meronym holonym derivation*	<semantic_ relation>	hypernym antonym meronym holonym derivation etc.
<collocation>	*hyponym*	<domain>	domain
<multiword>	*hyponym*		
<label>	domain		

Table 3: List of elements in each resource used for mapping and the type of relations they contain (if they are expressed implicitly, they are given in italics)

4.3 Mapping procedure

Because the structure of lexical entries are different in the two resources and because the semantically related words are not always encoded explicitly, some pre-processing steps were required before word senses in the two resources could be compared and mapped.

First, we extracted all the textual information from the elements we selected for the mapping procedure and if the elements contained free text (e.g. definitions), we performed part-of-speech tagging and lemmatization with ToTaLe (Erjavec et al. 2010), after which we filtered out all the function words so that only lemmas of nouns, verbs, adjectives and adverbs remained. This step was necessary because Slovene is a highly inflecting language.

Because sloWNet currently contains very few Slovene definitions and we believe that the semantically related words which are used in definitions would be very useful for mapping, we translated the English definitions for the synsets included in the sample with GoogleTranslate[1] into Slovene and then POS-tagged, lemmatized and filtered them in the same way as we did the Slovene definitions from SLD. In addition, we mapped the domain SLD labels to those used in sloWNet. If more than one sloWNet domain was possible for a SLD label, we used all of them.

We then created a so-called context vector for each sense of the word from the mapping sample for each resource, which contained all the words from the mapping elements. Collocations and multi-word combinations from SLD and semantically related literals from sloWNet were used in their canonical form whereas only lemmas of content words were used for the elements containing free text (i.e. definitions).

[1] http://translate.google.com/

SLD SENSE: N-jajce-1.1	SWN SENSE:eng-30-07840804-n
DEF: beljak 1 DEF: celica 1 DEF: hrana 1 DEF: prehrana 1 DEF: rumenjak 1 DEF: spolen 1 DEF: uporabljati 1 DEF: vsebovati 1 DOM: gastronomy 1 DOM: food 1	DEF: hrana 1 DEF: kokoš 1 DEF: ovalen 1 DEF: reproduktiven 1 DEF: telo 1 DEF: uporabljati 1 DEF: žival 1 DOM:gastronomy 1 ILR-hypernym: hrana 1 ILR-hypernym izdelek 1 ILR-hypernym prehrambeni izdelek 1 SYN: jajce 1

Figure 3: An example of context vectors of the same sense in both resources

An example of a context vector for one of the senses of the noun *jajce* (*egg*) from both resources is given in Figure 3. The information about the element in which the context word was found is retained together with its frequency.In this initial experiment the source information is only used for easier analysis of the mapping results but we plan to refine the mapping process by including this information as well.

Finally, the context vector of each SLD sense of a sample word of the same part of speech was compared to the context vectors of all of its sloWNet senses with the Simplified Lesk Algorithm (Kilgarriff and Rosenzweig, 2000) and the one with the highest lexical overlap was selected. If more sloWNet senses achieved the highest score, all of them were mapped to the given SLD sense. Similarly, the same sloWNet sense could be mapped to more than one SLD sense.

SLD		SWN		Analysis	
Entry	Sense	Synset ID	Synonyms/Definition	Matching words	Evaluation
jezik (language, tongue)	1. organ	eng-30-05301072-n	clapper, glossa, lingua, tongue *a mobile mass of muscular tissue covered with mucous membrane and located in the oral cavity*	jezik, organ, usten, votlina	OK
	1.1 food	eng-30-07652995-n	tongue / *the tongue of certain animals used as meat*	meso, organ, uporabljati, žival	OK
	2. means of communication	eng-30-05808557-n	language, linguistic process *the cognitive processes involved in producing and understanding linguistic communication*	jezik, razumevanje	CLOSE
	2.1 mental faculty			jezik, razumevanje	OK
	2.2. communication process			jezik, proces	OK
	3. way of expressing	eng-30-06282651-n	language, linguistic communication / *a systematic means of communicating by the use of sounds or conventional symbols*	način	OK
		eng-30-07082198-n	tongue / *a manner of speaking*	način	CLOSE
	3.1 in speech	eng-30-07082198-n	tongue / *a manner of speaking*	govor	CLOSE
		eng-30-05650820-n	language, speech / *the mental faculty or power of vocal communication*	govor	OK
	4. computer	/no mapping/	/no mapping/	/	/
	5. geography	eng-30-09442595-n	spit, tongue / *a narrow strip of land that juts out into the sea*	geografija	OK
	6. shoe part	eng-30-04450994-n	tongue / *the flap of material under the laces of a shoe or boot*	čevelj	OK

Table 5: An example of mapping results between SLD and SWN for the word *jezik* (Eng. *language, tongue*)

5. Results and discussion

5.1 Evaluation of the results

In order to evaluate the mapping procedure we manually checked the mappings for all the senses of the words in the mapping sample. In the evaluation, we used three labels for the suggested mappings:

- OK if the mapping was completely correct;
- CLOSE if the mapping was almost correct but slightly more general or more specific; and
- WRONG if the mapping was incorrect.

	N	A	V
No. of senses in SLD	61	62	75
No. of senses in SWN	57	52	58
% mapped SLD senses	78.7	96.7	96.0
% SWN mapped senses	70.2	80.8	60.3
% correct mappings	68.3	35.9	25.4
% close mappings	17.5	16.2	23.2
% incorrect mappings	14.3	47.9	51.4

Table 4: Manual evaluation of mapping results

As can be seen from Table 4, we were able to map over 90% SLD senses and nearly 80% sloWNet synsets, suggesting there is substantial overlap between the two resources despite the fact that the contexts of word senses are quite sparse. Accuracy is by far the highest for nouns (68.3%), which is not surprising because they are the easiest and most language-independent category and are therefore organized in a very similar way in both resources but probably also because sloWNet still has a much better coverage for nouns than for other words.

Verbs, which are a known to be a very difficult category in lexical semantics, perform the worst. They are also very language-specific, especially when the linguistic systems are as different as English and Slovene. What is more, sloWNet and SLD have quite different theoretical foundations, which is why they treat verbs very differently, making the mapping between their verbal senses even harder if not downright impossible in some cases.

In our experiment, verbs have the highest number of mappings that were evaluated as "close". These cases show that the mapping is in the right direction but in the case of more abstract verbs or verbs with highly dispersed and metaphoric usage, semantic tendencies are highly dependent on the concrete communicative situations and the related semantic descriptions. It is interesting to note that there are only 3% of SLD nominal senses that did not obtain a single correct or close mapping. The figure goes up to 25.6% for adjectives and 30.6% for verbs.

An example of mapping results for the noun *jezik* (Eng. *language, tongue*) are given in Table 5 where the senses from SLD are displayed along with their SWN mappings, lists of overlapping words and an evaluation tag of the accuracy of the mapping. In most cases, a sense from SLD was mapped to a single SWN sense (e.g. *organ*). But in some cases more than one SLD sense were mapped to the same SWN sense (e.g. *means of communication, mental faculty and communication process*), or vice versa (e.g. *way of expressing*).

When analysing which types of semantic relations contribute the most to successful mapping, we observe that those are: (near) synonyms, hypernyms, holonyms and domains. Hyponyms, which are explicitly encoded in sloWNet and appear among the SLD collocations and multiword combinations are frequent but not overlapping in many cases. While this is bad for the mapping process itself, it is extremely useful after the mapping has been completed because both resources can benefit from the complementary information.

5.2 Error analysis

When taking a closer look at the discrepancies in the mappings, we observed that wrong mapping could be the result of a sense missing from sloWNet, for example because it has not yet been translated from the Princeton WordNet or because it is language-specific. In total, there were 18 nominal synsets that were missing in sloWNet, 8 of which were language-specific and could only be added to sloWNet by deviating from the Princeton WordNet structure. There were 17 such adjectival senses, only 2 of which were language-specific, and 22 verbal ones, where as many as 15 are due to the differences between the linguistic systems.

In some cases senses only appear to be missing in sloWNet because they exist under a different expression or part of speech (e.g. *konj-horse* in the sense of *unit of measurement* that is found under *horse power-konjska moč* in sloWNet). On the other hand, sloWNet also contains some, but not many, senses which are not present in SLD because they are known in Slovene but were not attested in the Gigafida corpus and Word sketches or because they too are language-specific and only exist in sloWNet because they were translated from English. There were 4 such cases among the examined nominal synsets and 6 adjectival and verbal ones.

	N	A	V
No. of senses missing in SWN	18	17	22
No. of language-specific senses in SLD	8	2	15
No. of language-specific senses in SWN	4	6	6
No. of identical senses with no No lexical overlap	1	7	4

Table 6: Analysis of the mapping discrepancies

There were also a few very interesting cases in which both resources contained parallel senses that could easily be mapped manually by looking at the lexico-semantic information provided by the two databases. But since they do not contain identical words in the fields we used for mapping, the lexical overlap score is 0 and therefore could not be mapped automatically with the procedure we are using.

6. Conclusion

We have described a preliminary experiment in which we mapped senses of polysemous nouns, adjectives and verbs in two different kinds of semantic lexicons for Slovene where one has been developed manually, is corpus-based and is primarily intended as a dictionary resource for human users, and the other has been automatically translated from English and aims to enhance automatic semantic text processing. The mapping was based on the semantically related words the two resources have in common, only that they are encoded explicitly in one resource and used implicitly in the other.

The goal of the experiment was to establish whether and in what way the two resources are compatible and what is the impact of combining the approaches based on a foreign language resource on the one hand and on real Slovene data on the other. Even though the information on semantic relations was quite scarce for most lexicon entries, the mapping was efficient for a lot of the senses included in our sample, especially for nouns, concrete words and clearly delimited senses. We were less successful with adjectives and verbs that have very different organization in the two resources and still contain a lot of noise in the automatically constructed sloWNet.

The benefits of the mapping are threefold: (1) SLD has been enriched with lexical and semantic information and the ontological-semantic network structure it had been missing, and it has been turned into a multilingual resource via the intra-lingual links among synsets in wordnets for various languages; (2) sloWNet has been enriched with semantic frames, lexico-syntactic patterns, collocations, multiword units and usage examples which are very expensive to encode from scratch but make the computational lexicon much more valuable; and (3) the mapping has been an indirect proof that a sense inventory constructed based on a foreign language has excellent coverage of the senses that are relevant for Slovene with very few foreign concepts and is comparable to a large extent to a language-independent corpus-based lexical inventory of a similar kind, despite the heavy criticism of the approach in the linguistic community.

It is also important to consider that more general implications for similar tasks can be inferred from the mapping of two conceptually different resources. SLD sense distribution relies heavily on corpus data and syntactic patterns found in real texts, thus investigating primarily syntagmatic aspects of semantic relations. On the other hand, sloWNet as a database with sets of cognitive synonyms expressing a distinct concept largely ignores corpus evidence and syntagmatic patterns in which its literals are used. In this respect, results of this investigation bring a more general estimation of the relation between a more syntagmatic and a more cognitive approach to sense distribution. The described task can be seen as a source of future investigation of pattern-based monolingual (in the sense, described in Hanks 2007) vs. ontology-based multilingually-oriented semantic relations.

In the future we plan to extend the approach and perform a large-scale mapping of the entire databases. We will also be working on the refinement of the mapping procedure by including more semantically-related content in the context vector, which will be obtained from sloWNet's second- and third-degree semantic relations and machine-translated English usage examples. On a practical note, we wish to test whether the semantic-ontological structure that SLD inherited from sloWNet works well in it.

7. Acknowledgements

The work described in this paper has been funded the and by the Slovene national postdoctoral grant (Z6-3668) and the "Communication in Slovene" Project financed by the European Union, the European Social Fund, and the Ministry of Education, Science, Culture and Sports of the Republic of Slovenia.

8. References

Baker, C.F., Fillmore, Ch., Cronin, B. (2003). The Structure of the Framenet Database. *International Journal of Lexicography* 16/3, pp. 281-296.

Barnbrook, G. (2002). *Defining Language: A Local Grammar of Definition Sentences*. Studies in Corpus Linguistics: John Benjamins Publishing Company.

Burgun A, Bodenreider O. (2001). Mapping the UMLS semantic network into general ontologies. In: *Proceedings of AMIA Symp 2001:81–5*.

Dyvik, H. (1998). Translations as semantic mirrors. In: *Proceedings of Workshop W13: Multilinguality in the lexicon II of the 13th biennial European Conference on Artyificial Intelligence*, ECAI 1998, Brighton, Great Britain, pp. 24-44.

Erjavec, T., Fišer, D. (2006). Building the Slovene Wordnet: first steps, first problems. *Proc. of the Third International WordNet Conference* (GWA'06), Jeju Island, Korea, January 22-26, 2006.

Erjavec, T., Fišer, D., Krek, S. and Ledinek, N. (2010). The JOS linguistically tagged corpus of Slovene. *Proc. of 7th International Conference on Language Resources and Evaluation* (LREC'10), Malta, May 17-23.

Fellbaum, Ch. (1998). WordNet: *An Electronic Lexical Database*. Cambridge, MA: MIT Press.

Fillmore, Ch., Atkins, S. (1992). Towards a frame-based organization of the lexicon: the semantics of RISK and its neighbors". In: Lehrer, A., Kittay, E. (eds.): *Frames, Fields, and Contrasts*. Lawrence Erlbaum Associates: Hillsdale, New Jersey, pp. 75-102.

Fišer, D. (2007). Leveraging parallel corpora and existing wordnets for automatic construction of the Slovene wordnet. *Proc. of the 3rd Language & Technology Conference*, October 5-7, 2007, Poznań, Poland, pp. 162-166.

Fišer, D., Novak, J. (2011). Visualizing sloWNet. *Proc. of Electronic lexicography in the 21st century: new applications for new users (eLex'11),* Bled, 10-12 November 2011, pp. 76-82.

Fišer, D., Sagot, B. (2008). Combining multiple resources to build reliable wordnets. *Text, Speech and Dialogue* (LNCS 2546). Berlin; Heidelberg: Springer, 2008, pp. 61-68.

Gangemi, A., Guarino, N., Masolo, C. and Oltramari, A. (2003). Sweeting WordNet with DOLCE. *AI Magazine,* 24(3), 2003.

Gantar, P., Krek, S. (2011). Slovene Lexical Database. In: D. Majchráková, R. Garabik (eds.) *Natural language Processing, Multylinguality.* Sixth International Conference. Modra, Slovakia, 20-21 October 2011. Slovenská akademiá vied, Jazykovedný ústav L'udovita Štúra, pp. 72-80.

Green, R., Pearl, L., Dorr, B.J., Resnik, P. (2001). Mapping Lexical Entries in a Verbs Database to WordNet Senses. In: *Proceeding of ACL-EACL-2001*, Toulouse, France, 2001.

Hanks, P. (2004). Corpus Pattern Analysis. In: Williams, G., Vessier, S. (eds.) EURALEX 2004. *Proceedings*. Lorient: Université de Bretagne-Sud.

Kilgarriff, A., Krek, S. (2006). Slovene Word Sketches. *Jezikovne tehnologije, Institut " Jožef Stefan",* Ljubljana.

Kilgarriff, A., Rosenzweig, J. (2000). English SENSEVAL: Report and Results. In: *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, LREC, Athens, Greece.

Kilgarriff, A., Tugwell, D. (2001). WORD SKETCH: Extraction and Display of Significant Collocations for Lexicography. *Proceedings of the AVL workshop on COLLOCATION: Computational Extraction, analysis and Exploitation*. Toulouse, pp. 28-32.

Kosem, I., Husak, M., McCharty, D. (2011). GDEX for Slovene. *Proc. of Electronic lexicography in the 21st century: new applications for new users (eLex'11),* Bled, 10-12 November 2011, pp. 151-159.

Krstev, C., Pavlović-Lažetić G., Vitas, D. and Obradović, I. (2004). Using textual resources in developing Serbian wordnet. *Romanian Journal of Information Science and Technology*. 7/1-2, pp. 147-161.

Logar Berginc, N., Krek, S. (2010). New Slovene corpora within the Communication in Slovene project. In: *International Conference SLAVICORP. Corpora of Slavic Languages*. Dubrovnik, Croatia, November 22-24 2010. Abstract.

Loper, E., Szu ting Yi and Palmer, M. (2007). Combining lexical resources: Mapping between propbank and verbnet. In *Proceedings of the IWCS-7.*

Niles, I. and A. Pease (2003). Linking lexicons and ontologies: Mapping wordnet to the suggested upper merged ontology. In *Proceedings of the 2003 International Conference on Information and Knowledge Engineering (IKE 03)*, Las Vegas, Nevada, June 23-26 2003.

Reed, S. L. and Lenat, D. B (2002). Mapping ontologies into Cyc. In: *Proceedings AAAI Conference 2002 Workshop on Ontologies for the Semantic Web,* Edmonton, Canada.

Sagot, B., Fišer, D. (2012). Automatic Extension of WOLF. *Proc. of the 6th International Global Wordnet Conference* (GWC'12), Matsue, Japan January, 9-13, 2012.

Suchanek, F. M., Kasneci, G. and Weikum, G. (2008). YAGO - A Large Ontol- ogy from Wikipedia and WordNet. *Elsevier Journal of Web Semantics*, 6(3):203–217, September 2008.

Tonelli, S. and Pianta, E. (2009). A novel approach to mapping FrameNet lexical units to WordNet synsets. In *Proceedings of IWCS-8,* Tilburg, The Netherland.

Tufis, D., Cristea, D. and Stamou, S. (2000). BalkaNet: Aims, Methods, Results and Perspectives. A General Overview. In: Dascalu, Dan (eds.): *Romanian Journal of Information Science and Technology*. Special Issue. 7/1-2, pp. 9-43.

Evaluating Scope for Labeling Nominal Compounds using Ontology

Sruti Rallapalli, Soma Paul
Language Technologies Research Center, IIIT-Hyderabad
Hyderabad, 500032, India
sruti@students.iiit.ac.in, soma@iiit.ac.in

Abstract

Labeling nominal compounds with semantic relations is a challenging NLP task, as it requires the extraction of the hidden relation between the constituents of the nominal compound. In this paper, we explore the scope of identifying the semantic relation and thereby interpreting a nominal compound using an indexed, semantic ontology. This method has the following advantages over other approaches that use unstructured documents and classification models for nominal compound interpretation: 1. A semantic relation is much less ambiguous than a verb or preposition paraphrase. 2. Processing of an unstructured database is avoided. 3 Instances of infrequent nominal compounds are easier to handle, as there are no statistical predictions involved. However, one issue with our proposed system is the lack of robustness which arises due to the difficulty involved in obtaining a huge, generic ontology. This issue is addressed in our work by combining the ontology search with noun similarity measurement techniques to handle the cases that are not covered in our ontology.

Keywords: Nominal compound interpretation, Ontology, Semantic relations

1. Introduction

Compounding of nouns is a popular linguistic phenomenon occurring in many languages with varying flexibility and frequency, and has always been a topic of interest in NLP. About 3.9% of the words in Reuters are bigram nominal compounds (Baldwin and Tanaka, 2004). These compounds stand as remarkable cases of encrypted linguistic information as they sometimes encode a hidden semantic relation and meaning. The huge prevalence of nominal compounds (NCs) in literature and the existence of an implicit meaning and semantic relation between the combining constituents make NC interpretation an interesting yet challenging task in the NLP area.

Consider the example of *Lemongrass Oil*, which is composed of the nouns *Lemongrass* and *Oil*. The relation between *Lemongrass* and *Oil* is not explicated on the surface, although a user of the language can decode the implicit information correctly. WordNet (Fellbaum, 1998) for example states that *Lemongrass Oil* is an aromatic oil that smells like lemon and is widely used in Asian cooking and in perfumes and medicines. However, this information is not sufficient to paraphrase *Lemongrass Oil* as "oil extracted from Lemongrass" or to conclude that *Lemongrass* and *Oil* are related via the *Source* relation. Moreover, WordNet and most of the other existing knowledge bases randomly capture information about NCs. For example,WordNet has an entry for *fruit juice* but lacks mention of many other commonly used NCs such as *fruit cake, fruit pulp, fruit skin, fruit slices,* and *fruit bread*. On the other hand, ConceptNet (Havasi, 2007) is a rich semantic database containing concepts and relations between them, but fails to capture many of these NCs and the relation that exists between their constituent nouns. For example, ConceptNet contains the following information for the compound *Lemongrass Oil:* receives the action *extract from herb.* This information does not capture the relation that exists between the constituent nouns *Lemongrass* and *Oil*.

A literature survey shows that most of the approaches adopted for NC interpretation can be broadly classified into one of the two classes: (a) Supervised machine learning approach (b) Unsupervised data-driven approach. However, these approaches are not efficient enough to handle the sparseness of data which is a major issue in case of NCs. Most of these approaches collect and use statistics on the occurrence frequency of an NC. Thus, when an NC is rare and infrequent, which is mostly the case; the estimated probabilities become unreliable and lead to wrong interpretations.

An ontology-based approach overcomes this issue easily, as frequencies of data become irrelevant in the context of the ontology. Also, ontology being a structured database, search can always be accomplished in a more controlled way than in an unstructured database such as a corpus.One issue however, with the use of an ontology for the NC interpretation task is that it is hard to build an exhaustive generic ontology. Therefore the proposed system cannot be robust if only an ontology is used for relation extraction. We propose to handle this issue by adopting a hybrid approach that combines the use of an ontology with noun similarity measurement techniques, to handle those NCs which are beyond the scope of the ontology.

This paper is organized into the following sections. The Next Section gives a brief overview of the different approaches that have been proposed so far for handling the task of NC interpretation. Section 3 describes the architecture of the ontology in PurposeNet (Kiran Mayee et al., 2008), which has been used in our work as the semantic knowledge base for extracting semantic relations for NCs. The schema for representing the information pertinent to deducing semantic labels for NCs is discussed in Section 4. Section 5 presents a detailed analysis of the various types of NCs, which motivates the algorithm for extracting semantic relations for NCs from the ontology. The algorithm is explained in detail in Section 6. Finally we summarize the discussion regarding the scope of using an ontology for extracting semantic relation for NCs.

2. Related Work

Most of the approaches to NC interpretation can be classified into the following two classes: (a) supervised Machine Learning techniques, and (b) unsupervised Data-

driven approaches. Approaches that follow (a) enlist an inventory of semantic relations and perform compound interpretation majorly as a classification task, assigning to every compound a unique class defined by one of these relations. These approaches focus on word sense disambiguation and lexical specialization (Girju et al., 2004) in the semantic noun hierarchy database - WordNet. Kim and Baldwin (2006) use a set of seed verbs for every type of semantic relation. They construct templates for each seed verb, associating it with appropriate grammatical relations to the head and the modifier. They map the verb tokens in sentences to a set of seed verbs using *WordNet::Similarity*. Finally they identify the corresponding relation for each of the seed verbs obtained from the mapping and select the best interpreting semantic relation using a trained classifier. Kim and Baldwin (2005) propose a simplistic example-based interpretation approach, in which they annotate a few set of examples using an inventory of relations, and apply lexical similarity of the testing NCs with their pre-tagged training NC instances, using WordNet. Ó Séaghdha (2007a) implements SVM classifier techniques using WordNet and co-occurrence vectors on a dataset labeled using a set of 11 semantic relations (Ó Séaghdha, 2007b). The system exhibits best performance using binary classifiers and a linear kernel.

The second type of approach is usually unsupervised and data-driven with an open inventory of relations. The earliest work using a corpus was done by Lauer (1995), where the inventory contained a set of 8 prepositions. He built a probabilistic model, which computed the probability for a particular preposition by using the counts of noun-preposition-noun paraphrases in the corpus, and predicted the most likely prepositional paraphrase based on these probabilities. Lapata and Keller (2004) show that for majority of the tasks, including NC interpretation, simple unsupervised models perform better, although not outperform the state-of-the-art systems, when the n-gram frequencies are obtained from web rather than a corpus. Butnariu and Veale (2008) use Google n-gram patterns to extract the relational possibilities of both the head and the modifier. The possible paraphrases for the NC are generated using these extracted corpus-based relations. The possible paraphrases are all then ranked based on their occurrence in the corpora, to predict the relation for the NC.

Most of the research in recent times has progressed towards solving the problem of NC interpretation using classification techniques, barely using any conceptual information pertaining to the constituents of the NC. However, there are approaches that use large ontologies such as the Generative Lexicon approach (Johnston and Busa, 1996). They use qualia structures to represent all the lexical items in their ontology, and use phrase structure schemata to represent the combination of nouns to form compounds. In turn, they understand and interpret compound forms with the help of these schemata. There are other approaches that use ontologies but are restricted to the news or bio-medical domain. Specia (2006) has used a hybrid approach which couples knowledge base information along with weakly supervised corpus based techniques, in the KmI news domain, for the purpose of Intranet Annotation. She uses the Kmi-basic-portal ontology to map linguistic tuples containing nouns and verbs to the corresponding classes in the ontology, using similarity techniques, and predicts the relation. Little of this work based on ontology has been extended to the interpretation of NCs.

3. Design of Ontology

PurposeNet is an artifact ontology in which artifacts are organized in a multiple inheritance hierarchy. The ontology is built in Web Ontology Language (OWL), which is a W3C standard for Semantic Web. All the assertions in the ontology are represented using a set of standard XML tags. The present work only uses artifacts of the hotel and food domains, which are subsumed under tourism. The Hotel and food ontologies in PurposeNet are manually built using content available in Wikipedia. Every artifact in the ontology is described in terms of two features: (a) descriptive features and (b) action features. Every artifact is also connected to other artifacts by one of the two relationships – *subtype* and *component*. There are 20 descriptive features identified to describe an artifact. Refer to Table1 for these features. There are 7 action features that describe the artifacts. They are shown in Table 2. Consider the following example of *Butter_Knife.* Its descriptive features are listed in Table 1. Its action feature *Purpose* contains: *Purpose some Cut_Butter*. Each of the action features is again specified in terms of a set of semantic roles. For example, participants involved in the purpose action of butter knife are the following: *Instrument: Butter_Knife, patient: Butter* and *Agent: Human.* These descriptive and action features together capture all kinds of information associated with an artifact.

4. Schema for Representing Nominal Compounds in Ontology

NCs are multiword linguistic expressions that convey a concept. In endocentric type of compounds, one of the constituents is the head and other nouns are modifiers that convey some property of the head. For example, let us consider the following cases: *cheese knife*, *plastic knife*, *garden knife* and *chef knife*. Each modifier signifies a different aspect; each aspect manifests into a perspective (Langacker, 1987) from which the instrument *knife* can be viewed. In an exocentric compound, none of the constituents is a head, as the whole expression has an external referent. There also exist copulative NCs which contain two heads such as *washer dryer*. Each of these NCs need not be represented as a node label in concept ontology all the time. We observe that such a representation obscures the semantic relation that exists between the constituent nouns. For example, the existence of *wheat bread* and *garlic bread* as subtypes of *bread* results in loss of information that the former is *made with wheat* and the latter *contains garlic*. In order to capture the correct and precise relation between the constituent nouns, we have adopted the following strategy in representing NCs in the ontology. Exocentric noun compounds such as *Hotel chains* and *gamma knife* are represented as a node label in the ontology. Yet another occasion when a complex concept can be stored as a multiword expression in the ontology is the *garden knife*, which is a knife used in the garden. Since the makeup and shape of such a knife is different from the ordinary *knife*, this artifact cannot inherit its descriptive features from its parent which is *knife*. Therefore, *garden knife* is represented as a compound expression in the ontology. For all other compositional endocentric NCs, concepts corresponding to the head and the modifier occur as

separate node labels. Thus, NCs such as *tomato soup*, *petrol car* and *mustard oil* are not represented as they are in the ontology. In the NC *tomato soup*, *tomato* and *soup* occupy their respective positions under Vegetable and Food ontology and are connected by the feature *Component*. Section 5 covers the method for searching the head and modifier concepts in the ontology and deducing the relation from the available information. Section 6 presents a survey of NCs that we have done in order to understand various relations that exist between constituent nouns.

Descriptive Features	**Possible Values**	**Butter_Knife**
Color	{black, white, green}	{any
Constitution	{metal, plastic, foam, rubber}	{Steel,metal}
Fluidity	{fluid, nonfluid}	{nonfluid}
Heaviness	{light_weight, moderate_weight, heavy_weight}	{light_weight}
Inertness	{inert, reactive, alkaline, acidic}	{inert}
Mobility	{mobile, immobile}	{immobile}
Oiliness	{oily, nonoily}	{nonoily}
Physical_State	{solid, liquid, gaseous}	{solid}
Shape	{cubical, cuboidal, cylindrical}	{flat}
Size	{big, small, huge}	{small}
Sliminess	{slimy, nonslimy}	{nonslimy}
Smell	{pleasant, unpleasant}	{no_smell}
Smoothness	{smooth, rough}	{smooth}
Softness	{soft, hard}	{hard}
Sound	{silent, soft_sound, bearable_sound, harsh_sound}	{silent}
Stability	{stable, nonstable}	{stable}
Subtleness	{subtle, nonsubtle}	{nonsubtle}
Taste	{sweet, sour, bitter}	{no_taste}
Temperature	{hot, cold, room_temperature}	{room_temperature}
Transparency	{transparent, translucent, opaque}	{opaque}
Viscidity	{viscous, nonviscous}	{nonviscous}

Table 1: Descriptive Features for *Butter Knife*

5. Analysis of Data

We use a simple extraction mechanism to extract artifacts from our ontology, create development data for identifying different semantic relations and for the classification experiment. The XML file of the ontology is used to extract all the 616 artifacts that are covered in the ontology. The NCs formed by each of these nouns are acquired from the Web IT corpus of Google n-grams using simple templates. We first extract trigrams containing our noun using the template <*> noun <*>. For example, the template <*> *Coffee* <*> resulted in 20755209 trigrams. Some of the trigrams and their counts are listed in table 3.

Malabar Coffee Beans	*76*
Manor Coffee Shop	*239*
Manual Coffee Grinder	*677*
Marble Coffee Tables	*1208*

Table 3: Google Trigrams and their counts

The obtained trigrams were then parsed by using the NLTK parser. All sequences of two noun words excluding those preceded or succeeded by a noun and those containing non-alphabetic characters were extracted. 430 NCs were randomly picked up from the resulting list of NCs to form a small set of development data, which was annotated with the semantic relations and used for checking the robustness of the classification system within the domain.
Numerous annotation schemes have been proposed by different people, each varying by the number of relations, and the level of abstraction. One annotation scheme which covers most of the possible compounds with the use of semantic relations, with clear boundaries and sufficient coverage of the different relation types is the state of the art inventory of 22 relations (Moldovan and Girju 2004; Girju 2006). This annotation scheme was used for annotating the NCs in our development data as most of the semantic relations enlisted in it such as Topic, Theme, Purpose, Property, Cause, Recipient, Hypernymy, Meronymy are captured in the features in our ontology.
Each NC was annotated with the most appropriate relation from the inventory. In case of ambiguity, an NC was annotated with more than one relation. For example: a *Cheese Knife* exhibits the *Purpose* relation, while a *Cheese Pizza* has a *Component* relation. Its paraphrase would be 'a pizza made of Cheese' or 'a pizza that contains Cheese'. Similarly, *Lemongrass Oil* has a *Source* relation, *Door Knob* has a *Part-Whole* relation, a *Coffee Machine* exhibits *the Purpose* relation, and a *Banana Fruit* exhibits *Hypernymy*. On the other hand, *Water Sprinklers* can have either a *Purpose* or *Theme* relation. Since there is no way to choose one over the other, the NC was annotated with both the relations. A *Dining Room* has both *Location* and *Purpose* relations. Similarly, *Door Curtains* shows both *Location* and *Purpose* relations. Most of the ambiguous cases contain the relation *Source* and *Component*, or *Purpose* and *Component*, or *Purpose* and *Location*. Examples of these cases are: *Corn Flakes*, *Metal Cutter* and *Floor Lamps* respectively. Less than 5% of our data contains such ambiguous cases. The distribution of NCs among different classes is shown in Table 4. The development data was made semantically rich with NCs from different classes. Some of these NCs when given as input to the ontology search system remained unpredicted, while some were beyond the scope of our system. One assumption on which our semantic interpretation system runs is that the ontology is complete and has good

Action Feature	Subtypes	Definition	Some Values for Car
Birth		Manufacture of artifact	Fix_Chassis_to_Body, Attach_Seats, Attach_Tyres
Purpose		Purpose of artifact	Transport_Human
Maintenance	General_Maintenance	Maintenance of artifact	Clean_Car, Clean_Engine
	Repair_Maintenance		Repair_Car, Repair_Engine
Wear and Tear		Wear and tear of artifact	Burst_Tyre, Overheat_Engine
ProcessRel		Actions the artifact can perform	Board_Passengers, Move_from_A_to_B, Alight_Passengers
Set up	First time Set up	Set up the artifact for functioning	Check_Ignition_System, Check_Brake
	General Set up		Check_Tyre, Check_Brake
Result_On_Destruction		Results on destruction of artifact	Engine – recycled to metal, Seats - resused

Table 2: Action features for Car

coverage of all concepts within the domain. With this assumption, when the unpredicted cases were studied, it was found that out of the 78 unpredicted cases, 20 of them are copulative, like, *Cream Cheese, Pool Area, Phone Cover, Coffee Table,* and *Tea Sandwiches.* In each of these examples, both the constituents of the NC are artifacts in our ontology.

Example: *Coffee Table* is made up of the nouns *Coffee* and *Table*. Both these nouns are artifacts but the relation between the two is not captured by the descriptive or action features of *Coffee* or *Table.* Such cases therefore remain unpredicted. We also found about 30 cases of NCs in which the modifier is an artifact in our ontology, but our system fails to interpret the NC, as in the case of *Cutlery Set, Cheese Ball, Milk Prices, Hotel Chain, Bar Light and Curtain Accessories.* This is because of the difficulty in capturing the information that a *Cheese Ball* is a *Ball made from Cheese* or a *Ball made up of Cheese.* The ontology instead may capture the ball as an artifact whose *Constitution* is *plastic* or *metal.* Other such NCs are *Ice Cream, Main Course and Hotel Accommodation*.

Relation	**Count**
Purpose	126
Part-Whole(Meronymy)	46
Is-A(Hypernymy)	33
Source	45
Theme	24
Property	22
Location	14
Component	97

Table 4: Distribution of NCs among different semantic relations

All the unpredictable cases were manually annotated with one of the relations from the inventory. It is found that most of the unpredictable NCs exhibit the *Purpose* relation. The other often unpredictable relations are *Location* and *Part-Whole*. The distribution of NCs among the various relation classes is shown in Table 5.

Relation	**Count**
Purpose	24
Part-Whole(Meronymy)	14
Theme	4
Is-A(Hypernymy)	6
Source	7
Topic	4
Property	9
Location	10

Table 5: Distribution of unpredicted NCs among different semantic relations.

6. Methodology

The basic strategy that we implement for NC interpretation is given below:

-Given an NC, we locate its head and modifier in our ontology, and extract their corresponding descriptive and action features (together referred to as features, hereafter). We then extract or predict the feature that connects the head and the modifier.

-To extract the features of the head and modifier in an efficient way, we index the nodes in the ontology. Then, given a head or a modifier, we obtain its corresponding index, and acquire all its features by traversing from the root node of the ontology tree to the node containing the required artifact in a top-down manner. The indexing will help in traversing from a given node to its ancestors or descendants.

-Once the features are acquired for both the constituents, the system adopts different search mechanisms to find the suitable semantic relation.

6.1 Indexing of Ontology

Indexing the ontology is important for quick accessing of the nodes and for easy traversal to the ancestors and descendants of the node. While there are many indexing mechanisms based on description logic and the inherent OWL indexing, they do not provide help in fast and easy acquisition of information or features of a given node. We therefore adopt the Dewey Encoding scheme

to index the nodes in our ontology tree. This scheme of indexing also helps us in finding the path between any two nodes in the ontology, which is well required for our search algorithm.
The root node of the ontology is *Entity* and it is assigned the '0' index. *Entity* is further classified into *Abstract_Entity* and *Physical_Entity*, whose indices are 0.0 and 0.1 respectively. *Abstract_Entity* has a subtype, *Action* (0.0.0), containing all the action features defined in the ontology while *Physical_Entity* has a subtype, *Artifact* (0.1.0), containing all the 616 artifacts that were manually added. Each of the remaining nodes is assigned a Dewey Encoded index that gives the absolute path of the node from the root. For example, *Food* is given the index 0.1.0.7, while *Liquid_Food*, which is a subtype of *Food*, is given 0.1.0.7.0. We maintain an index table containing names of the nodes and their corresponding indices as shown in table 6.

Coffee	0.1.0.7.5.8.0
Tea	0.1.0.7.5.8.1

Table 6: Sample Index table

To retrieve the features of a node such as *Coffee*, one must start from the node '0', collect all its features, proceed to its descendant '0.1', and collect its features. This must be repeated till the *Coffee* node is reached. While doing so, if a feature is re-defined in the child node, the value of that feature gets overridden with the newly defined value.

6.2 Search Algorithm

Different types of NCs require search mechanisms of varying complexity. We postulate below 4 different search traversals to handle the different types of NCs. They are: (a) One level Search (b) Multi level Family Search (c) Multi level Simple Search (d) Unique Node Search.
One level Search - The simplest case is when there is a direct relation between the head and the modifier. In such cases a single level search through the features of the head gives the relation. Consider *Lemon Tea. Tea* includes a feature *Component (Tea, Lemon)* as shown below. *S*o the system will predict the relation *Component.*

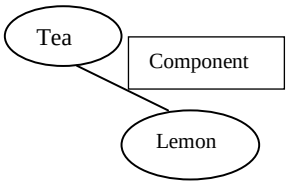


Figure 1: Feature representation for *Lemon Tea*

Multi level Family Search - In this case, the first search through the features of the head does not match the modifier. The next level search takes each of the above features and looks for the modifier in their family - parent, siblings or children. Consider *Mango Pickle. Pickle* contains a feature *Component (Fruit, Pickle) and Fruit* has a feature *Subtype (Fruit, Tomato)* as shown below. In such cases, when a *Subtype* relation is involved, the relation predicted in the previous level is retained. Here, the relation will be *Component.*

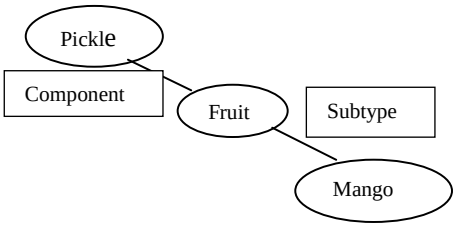


Figure 2: Feature representation for *Mango Pickle*

Multi level Simple Search - In this case, the first search through the features of the head contains the modifier as a substring. That feature is then retrieved, and the next level search for the modifier is performed over the features of this feature. Consider the case of *Bread_Toaster.* It contains a feature *Purpose (Bread_Toaster, Toast_Bread).* Extract *Toast_Bread* and perform the next level search on it. *Toast_Bread* contains *Patient (Toast_Bread, Bread)* as shown below. The feature in the first search is retrieved as the relation. Here it is *Purpose.*

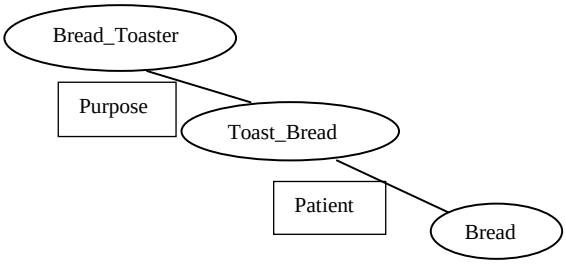


Figure 3: Feature representation for *Bread Toaster*

Unique Node Search - In this case, the unique NC is represented as a single node in the ontology. However, the search traversal for the unique node may be *One Level Search* or *Multi level Search,* as discussed above. Ex: *Ginger_Bread* contains a feature *Component (Ginger_Bread, Ginger)* and the relation can be retrieved by a *One level Search.* But *Hair_Conditioner* will require *Multi level Search. Hair_Conditioner* contains the feature *Purpose (Hair_Conditioner*,*Smoothen_hair)* and *Smoothen_hair* contains the feature *Recipient (Smoothen_hair, Hair)* as shown below. The relation here would be the feature obtained in the first search - *Purpose.*

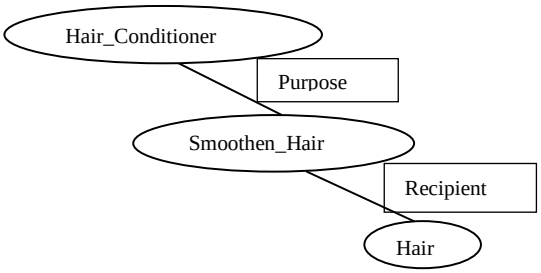


Figure 4: Feature representation *for Hair Conditioner*

When an NC is given for interpretation, there are many possibilities, which are postulated below.
(a)It can be unique and exist as a single node in the ontology.

(b)It can be non-unique. In this case only the head may be covered in the ontology, or only the modifier, or both head and modifier as different nodes.

Because no information regarding the uniqueness of an NC is available, we cannot choose a particular search traversal to apply. The system must therefore perform all the possible search traversals for a given NC and retrieve all the possible features that interpret the NC relation, without any ranking. These features are then mapped to their corresponding relations in the inventory using a map between PurposeNet features and inventory relations. These relations are produced by the system as the possible semantic relations between the constituents of the NC. This approach can predict semantic relations only when both the head and the modifier are included in the ontology, either as a single node or otherwise. In other cases where the modifier is not present in the ontology, or both the head and the modifier are not covered in the ontology, this algorithm cannot predict a semantic relation. Hence, we combine the ontology search approach with noun similarity techniques to increase the robustness of the system to handle compounds beyond the scope of the ontology.

(a)Consider the simple case when the modifier is not present in the ontology such as *Lemon Juice.* If the ontology contains Juice, but does not contain *Subtype (Fruit, Lemon),* then we need to calculate the semantic similarity for every feature of *Juice* with *Lemon.* We use a lexical similarity measure based on the distributional hypothesis, which extracts the context of both the lexical items, and if the lexical items have similar co-occurrence patterns, the two items are lexically similar. Here *Juice* has a feature *Component (Juice, Fruit),* and *Lemon* and *Fruit* will have similar co-occurrences with food, juice, tree, liquids, energy and so on. Thus, *Lemon* and *Fruit* are similar. We also calculate the similarity of *Lemon* with all other features, and choose the pair with maximum similarity. Then, our ontology search algorithm will be used to extract the semantic relation between the most similar pair, say, *Fruit* and *Juice.* The same relation will be predicted for our NC *Lemon Juice.*

(b)A more difficult case to handle is when neither of the two constituents of an NC is captured in the ontology. Consider *Tomato Soup*. If the ontology does not contain the node *Soup* or *Tomato,* we extract the head of the NC (*Tomato)* and check its similarity with each of the single nodes in our ontology. Then, we rank the pairs on the basis of their similarity scores, and choose the most similar pair, such as *Sauce.* Once we have a head in the ontology, we use the ontology search algorithm to extract the semantic relation between *Sauce* and *Tomato.* However, if the modifier is not present in the ontology, we follow the approach discussed in (a).

7. Conclusion and Future Work

The currently implemented ontology look-up method, if coupled with noun similarity techniques, promises good results for generic NCs. All the domain-specific NCs, as well as a great deal of other NCs will be predicted. However, the search algorithms implemented so far are basic. We plan to bring in a rule-based search, where the rules will depend on some ontological information of the constituents of the NC, like the distance between the nodes, uniqueness of the NC, its representation in the ontology (as distinct head and modifier nodes, or as a unique node) etc. We will compare the results with our current search algorithms, for both domain-specific and generic NCs. Further, the attributes in our ontology can be mapped to paraphrasing verbs in other languages. Then it will enable an English language NC to be paraphrased into other languages using a domain specific ontology containing English concepts.

8. References

Timothy Baldwin and Takaaki Tanaka. 2004. Translation by Machine of Complex Nominals: Getting it Right. In *Proceedings of the ACL 2004 Workshop on Multiword Expressions: Integrating Processing*, pages 24-31.

Cristina Butnariu and Tony Veale. 2008. A concept centered approach to noun-compound interpretation. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 81–88, Stroudsburg, PA, USA. Association for Computational Linguistics.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database.* MIT Press.

Roxana Girju. 2006. Out-of-context noun phrase semantic interpretation with cross-linguistic evidence. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 268–276, New York, NY, USA. ACM.

Catherine Havasi. 2007. Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge. In *the 22nd Conference on Artificial Intelligence.*

Michael Johnston and Frederica Busa. 1996. Qualia structure and the compositional interpretation of compounds. In *Proceedings of the ACL SIGLEX Workshop on Breadth and Depth of Semantic Lexicons*.

Su Nam Kim and Timothy Baldwin. 2005. Automatic interpretation of noun compounds using wordnet similarity. In *Proceedings of the 2nd International Joint onference on Natural Language Processing, Jeju Island, South Korea, 1113,* pages 945–956.

Su Nam Kim and Timothy Baldwin. 2006. Interpreting semantic relations in noun compounds via verb semantics. In *Proceedings of the ACL-06 Main Conference Poster Session.*

Ronald W. Langacker. 1987. *Foundations of cognitive grammar: Theoretical Prerequisites*. Stanford University Press, Stanford, CA. Vol 1, 1987(Hardcover), 1999(Paperback).

Mirella Lapata and Frank Keller. 2004. The web as a baseline: Evaluating the performance of unsupervised web based models for a range of nlp tasks. In *Proc. of Human Language Technologies - North American Chapter of the Association for Computational Linguistics (HLT NAACL*, pages 121–128.)

Mark Lauer. 1995. Designing statistical language learners: Experiments on noun compounds. Technical report.

Dan Moldovan, Adriana Badulescu, Marta Tatu, Daniel Antohe, and Roxana Girju. 2004. Models for the semantic classification of noun phrases. In *Proceedings of the HLT-NAACL 2004: Workshop on Computational Lexical Semantics*, pages 60–67, Boston, USA.

Diarmuid Ó Séaghdha. 2007a. Annotating and learning compound noun semantics. In *Proceedings of the ACL-07 Student Research Workshop*, Prague, Czech Republic.

Diarmuid Ó Séaghdha. 2007b. Designing and evaluating a semantic annotation scheme for compound nouns. In *Proceedings of the 4th Corpus Linguistics Conference*, Birmingham, UK.

Kiran Mayee P., Rajeev Sangal, Soma Paul, and Navjyoti Singh. 2008. An ontological resource organized around purpose. In *Proceedings of the 6th International Conference on Natural Language Processing*, Pune, India.

L. Specia, C. Baldassarre, and E. Motta. 2006. Relation extraction for semantic intranet annotations. Technical report, Knowledge Media Institute, Milton Keynes.

Simple Unsupervised Topic Discovery for Attribute Extraction in SEM Tasks using WordNet

Abhimanu Kumar, Richard Chatwin, Joydeep Ghosh

The University of Texas at Austin , Adchemy Inc., The University of Texas at Austin
Department of Computer Science, Adchemy Research Lab, Department of ECE
abhimanu@cs.utexas.edu, richard@adchemy.com, ghosh@ece.utexas.edu

Abstract

We present here a simple approach for topic discovery to extract attributes of online products using Wordnet. Identifying product attributes is important for search engine marketing (SEM) since it is integral to the ads displayed for search queries (Moran and Hunt, 2009). Our wordnet based model provides a simple, scalable and high precision attribute extraction mechanism. It is well suited for identifying attributes for previously unseen product categories and thus works specially well for SEM scenario. It outperforms unsupervised topic discovery approaches such as LDA for SEM tasks on 4 online product datasets. The model has been successfully implemented as a production version code for ad-copy creation.

1. Introduction

Information extraction has been an active area of research in Natural Language Processing. It is useful for obtaining query-able information databases from unstructured data such as webpages, news articles etc. Information extraction approaches has been applied to a variety of tasks from obtaining protein names from biological papers (Fukuda and Tamura, 1998) to building dictionaries (Riloff and Jones, 1998). These techniques have also been used to extract relationship among entities (Zelenko et al., 2003) and entity attribute extractions (Bellare and Talukdar, 2007) using training seed sets.

But all the work so far has focused about: a) finding entities when the entity types are known for ex: finding a person or location from a text, or b) extract entities/relations using a seed set to train the model. This can be problematic for entities hitherto unseen by the model. We propose a simple and scalable information extraction model to discover new entity types without any seed set or prior knowledge of the types to be extracted. This scenario is typically encountered in search engine ad-copy creation process where the attributes of a product being advertised can vary from one product subcategory to another. The seed labels are of not much use in this case. Our proposed model uses WordNet semantic similarity metrics to obtain product attribute sets. The input of the model is online product category catalog and the output is a set of clusters each representing an attribute of the product. This model is well suited for ad creation in SEM tasks and can also be used as a bootstrapping tool for general attribute extraction problems.

The model is unsupervised and doesn't need any seed set for training, though it uses WordNet semantic structure to find the attributes. This makes it highly scalable to newer product categories. The model outputs a specified number, κ, of topics or attribute clusters and ranks them in the decreasing order of confidence. Ad-copy creation process, described in section 2., needs to know the prominence of a product feature and whether to include it in the ad-display. The ranked output of the model helps here in deciding the relevance of an attribute for a given product category. We compare our model with traditional unsupervised topics discovery models such as LDA on 4 SEM datasets. It performs better than LDA on all 4 datasets as reported in later sections. Though the model works well for SEM related tasks and datasets, it is not a generic model like LDA (Blei et. al, 2003). It exploits the unique properties of an SEM task and corresponding datasets and is built for such a task. We discuss the cases when it might perform poorly.

Our attribute extraction model sits at the unique juncture of word-semantics, Ontology, and data statistics based extraction techniques. We combine WordNet based "sense-ontology" and semantic metrics with statistical information present in the data to discover relevant attribute-clusters.

2. Problem Definition

The primary focus of search engine marketing is displaying appropriate ads on search engines for a search term. SEM firms maintain a set of appropriate advertisements related to each search term and choose the best ad from this set based on certain relevance criteria. Table 1 shows a search term "Chaise Lounge" and the corresponding set of candidate ads to be shown. Producing this set of ads is one of the big challenges of SEM. These sets of candidate ads are short sentences made of essentially two parts: a) Intent and b) Noun Phrase. The intent of the ad tells the purpose of the ad, e.g. in "IKEA leather chaise lounge on sale", "on sale" is the intent, and in "buy cheap colorful furniture", "buy" is the intent. The noun phrase is the product being talked about in the ad. In "IKEA leather chaise lounge on sale", "IKEA leather chaise lounge" is the noun phrase, and in "buy cheap colorful furniture", "cheap colorful furniture" is the noun phrase. Noun phrase in the ad is a sequence of the product and its attributes, e.g. "IKEA leather chaise lounge" is made of "IKEA" + "leather" + "chaise lounge". Formally all this can be expressed in terms of a context free grammar as:

$$\begin{aligned} \langle ad \rangle &= (\langle intent \rangle)^{\star} \langle noun\ phrase \rangle (\langle intent \rangle)^{\star} \\ \langle noun\ phrase \rangle &= (\langle attribute \rangle)^{+} \langle product\ name \rangle \end{aligned} \quad (1)$$

Chaise Lounge		
IKEA leather chaise lounge on sale	affordable home furniture	buy cheap colorful furniture

Table 1: 3 candidate ads for search term "Chaise Lounge"

The $\langle intent \rangle$ is easy to obtain, but finding $\langle attribute \rangle$ set requires domain knowledge. The $\langle attribute \rangle$ of a product helps in defining the specificity of the ad by: a) targeting a specific set of consumers who are interested in that attribute, and b) providing information about the category of products which are available for that $\langle intent \rangle$ at the sellers facilities. E.g. the product toy can have several attributes and the ad "$wooden\ brain-teaser\ puzzles\ for\ sale\ at\ walmart$", with the help of "wooden" and "brain-teaser" attributes, targets the set of consumers who are interested in wooden brain-teaser puzzles. Knowing this attribute-cluster requires going through the toy catalog of the store and manually extracting these attribute-clusters. For ex: "wooden" attribute is a member of "material" attribute-cluster of toy.

Our model solves this problem by automatically extracting the set of attributes using the seller product catalog. It extracts the attributes as well as provides label to each attribute set. For the product toy mentioned above, the model discovers the attribute-clusters : $\{wooden, leather, plastic, tin \ldots\}$ and $\{red, green, blue, black \ldots\}$ and provides labels "$material$" and "$color$" respectively to these 2 clusters. This helps in ad-copy creation. The ad-copy creation is an extension of the ad-generation scheme in equation 1. The difference is in the $< noun\ phrase >$ generation where the new scheme is:

$$\langle noun\ phrase \rangle = (\langle attribute \rangle_1)?\ \ldots\ (\langle attribute \rangle_n)?\ \langle product\ name \rangle \quad (2)$$

In the ad-copy equation 2 above, the attributes of the product are assigned certain order to give the ad semantically correct structure. For ex: "coffee-colored women's t-shirt" is semantically/aesthetically better than "women's coffee-colored t-shirt". Knowing attribute-cluster labels makes obtaining the right order among attributes easy.

3. Related Work

A variety of approaches have been used from generative (Freitag and McCallum, 1999) and discriminatory schemes (Yu, Lam and Chen, 2009) to rule based models (Reiss and Raghavan, 2008). Entity and attribute extraction is an important subtask of Information Extraction problem.

Generative and Structure Learning based extraction. (Eisenstein and Yano, 2011) provide a non-parametric generative scheme for named entities extraction from text. It uses supervision from an initial set of 5 prototype examples. (Reisinger and Pasca, 2009) show that an LDA based generative scheme is the best approach for expanding WordNet hypernym-hyponym structure via attribute extraction.

Ontology based extraction. (Maedche et. al, 2003) provide an ontology based information extraction technique which uses weighted finite state machines. The approach is generic to information extraction tasks and does not specialize in attribute extraction as well the finite state machines need supervision. Moreover, they use German corpora for all the evaluation. (Embley et. al, 1998) use domain based ontologies for information extraction. After choosing the relevant ontology they formulate a set of rules for extracting constants and keywords.

Tag based supervised extraction. (Ghani et al., 2006) treat each product as attribute-value pair and use a set of seed labels to induce a classification setting for extraction. Their first step is to define a set of attributes to be extracted. (Putthividhya and Hu, 2011) provide a tag based brand name extraction technique for online products. Their problem overlaps with the SEM problem as ads need brand names too. They use ebay shoes and clothing product catalog as their corpus.

Semantics and Rule based extraction Nagy and Farkash (2010) assign webpages to people based on the attributes matched among them. They manually mark relevant attributes then formulate empirical rules to extract attribute values. (Nagy and Farkas, 2008) provides logic based approach to extracting class attributes from English texts. Etzioni (2005) et. al provide an experimental study with an extraction scheme for obtaining named entities from web. Their scheme relies on domain independent extraction patterns to generate candidate named entities.

All the above approaches can be classified into two categories: a) they use a seed set for training, or b) they use a pattern or rule empirically discovered for the extraction. Due to this fact, all of the above approaches are insufficient for our requirement because they are not scalable to hitherto unseen product categories. And a prior knowledge of what the attributes are is needed in all of the approaches. Our model deals with both of these issues through utilizing the semantic clustering of words based on WordNet metrics. It is completely unsupervised in terms of seed sets or rules/patterns. The approach that comes closest to solving the SEM problem is unsupervised topic model (Blei et. al, 2003) as this too doesn't need any seed set or assume any rules.

4. Document Collection

The datasets used for the models are product catalogs of different online product categories. This is done to make sure that the models face the same issue as in the real world SEM tasks. The real world SEM techniques use sellers' product catalog to generate relevant ad-copies. We use 4 different online product-catalogues of 4 different sellers: 1) Furniture Catalog, 2) Clothing Catalog, 3) Watches Catalog, and 4) Beddings Catalog. We compare our model with LDA and a baseline and report the results.

The model uses four datasets, two for parameter tuning and two for testing. All Four datasets are product catalogs of

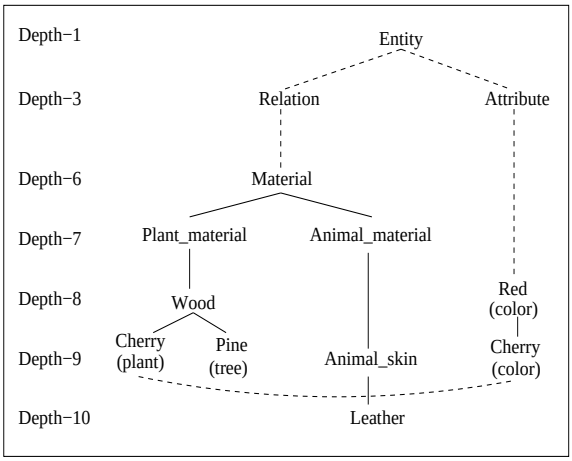


Figure 1: An example sense hierarchy for WordNet, nodes at the same height have the same depth in WordNet. The depth at each height is labeled in the left side of the figure. Immediate parents are connected with a solid line.

online products on sale. Each product entry in a catalog is one line.

Furniture Catalog. This furniture catalog has 7677 product entires and 49709 words. Each entry is a long phrasal noun eg. "Cambridge Black 25-Inch Backless Counter Swivel Stool with Black Vinyl Cushion Seat".

Clothing Catalog. This is a catalog of clothes with 16839 product entries and contains a total of 24485 words. A typical entry is: "Plus Size Full Bust".

Watches Catalog. This is a catalog of watches. It has 7982 entries and 68397 words. A typical entry looks like "Nixon Men's 'The Rocker' Stainless Steel and Leather Quartz Watch".

Beddings Catalog. This is Beddings catalog with a total of 22955 product entries and 153552 words. A typical entry here looks like "Frette Completo Letto Textured Queen Bedspread".

5. The Attribute Extraction Model

The model treats the product catalog as a bag of words. Each word w present in catalog d is assigned a probability mass $P(w|d)$ as follows:

$$P(w|d) = \frac{N_d(w)}{\sum_{\forall w \in d} N_d(w)} \quad (3)$$

where $N_d(w)$ is the count of word w in catalog d. The sense-set ψ_w for each word w is the set of all senses of w, i.e.

$$\psi_w = \{w_s : w_s \in synset(w)\} \quad (4)$$

where $synset(w)$ contains the SynSets of word w in WordNet (Miller, 1995). Each member w_s of set ψ_w is a unique sense of word w and lies in a unique SynSet of w. The catalog d is expanded to a "bag of senses", Ψ, where:

$$\Psi = \cup_{\forall w \in d} \psi_w \quad (5)$$

A naive approach to clusters these senses is to group two senses, w_{s_1} and w_{s_2} , together if $hypernym(w_{s_1}) = hypernym(w_{s_2})$, i.e. w_{s_1} and w_{s_2} are immediate siblings in WordNet hypernym tree. Figure 1 shows an example where two immediate sense siblings, "cherry" and "pine" are clustered using a common parent "wood". "wood" becomes the cluster head of this cluster. This approach can be extended to include "leather" in the cluster with "material" as the new cluster head. "material" is a valid cluster label and we propose later in this section a model that arrives at such valid cluster heads or labels.

5.1. Modeling

Aforementioned naive approach of clustering words based on WordNet sense hierarchy does not know how to arrive at valid cluster heads, i.e. when to stop adding more hierarchies to the sense tree . This task can be achieved by utilizing 2 important semantic metrics:

- **depth-metric**: the sense of a word increases in specificity as the word's depth increases in WordNet (Jiang and Conrath, 1997)
- **hop-metric**: the smaller the hop-counts between two words in the WordNet taxonomy the closer their senses are (Rada et al., 1989).

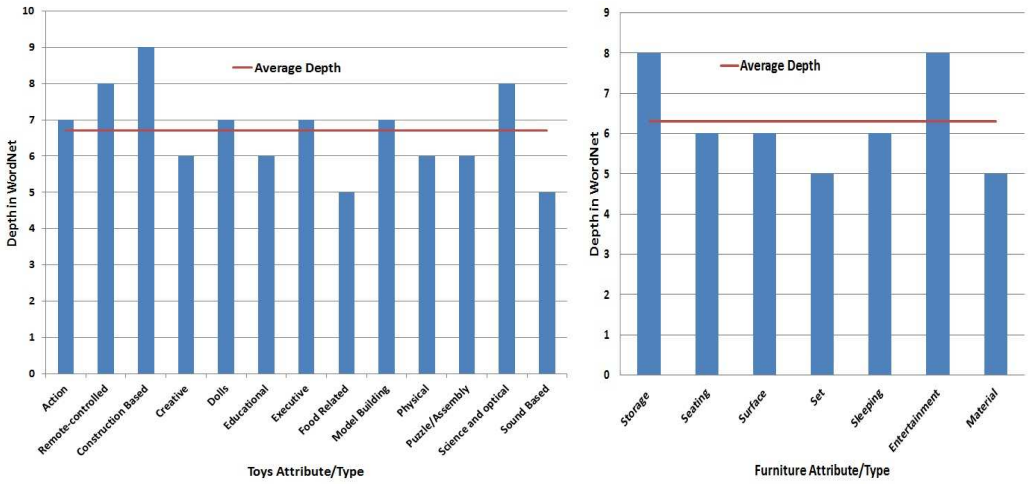
The proposed model tunes its parameters based on the above 2 metrics. The model learning has 2 phases: 1) Cluster Discovery, and 2) Cluster Pruning.

5.2. Cluster Discovery (Phase I)

Algorithm 1 describes the cluster discovery process in detail. The model iterates through each word-sense present in the "bag of senses", Ψ obtained from equation 5, and clusters them together based on WordNet's hyponym-hypernym (IS-A) relation. For each sense $s \in \Psi$, the algorithm first iterates through all the discovered clusters in cluster set Ω and checks whether $\exists C_i \in \Omega \ni s$ has a valid hypernym/hyponym relation with C_i. If $\exists C_i \in \Omega$ then s is added to C_i and $C_{i_{head}}$ is modified appropriately. If there is no such C_i then the model iterates through the hitherto unclustered senses $s_j \in \Psi$ such that s and s_j has a hypernym/hyponym relationship between them. If there exists such an s_j then a new cluster C_{new} is created with s and s_j inserted into C_{new} and $c_{new_{head}}$ appropriately initialized. This C_{new} is inserted into hitherto discovered cluster Ω. The model moves onto the next sense in Ψ and starts the above steps again. After iterating through all elements of Ψ, Ω returns with a set of candidate clusters/topics with each cluster's head assigned as label for that topic. The labels of these clusters will become our discovered attributes of the product. Each $C_i \in \Omega$ is a cluster of word-senses with a sense-hierarchy among the elements present in it.

5.3. Cluster Pruning (Phase II)

The clusters obtained in phase I contain lot of noise and are not sense specific. Table 2 shows some of the prominent clusters discovered after phase I in the Furniture Catalog. The Cluster Pruning phase deals with by parametrizing the cluster properties based on the WordNet metric defined in section 5.1.. The clusters have the following properties:



(a) Toys attributes depth

(b) Furniture attributes depth

Figure 2: WordNet depths of various attributes of Toys and Furniture catagory obtained from sample Wikipedia pages

unit	set	play	event	material	furniture	equipment	wood	colour	leather
unit	set	play	event	stuff	dresser	equipment	wood	color	leather
clark	quartet	Hole	turn	Soda	chest	Bird	Ash	Red	Kids
london	suite	set	label	Lime	bureau	glove	Hardwood	White	Buff
almond	product	Sets	carry	Products	Table	hammer	Wicker	Yellow	Suede
hamilton	Core		Case	Crib	Etagere	Set	Alder	Jade	Micro-suede
clock	Triplet		Pawn	Stool	Sleeper	Bag	Birch	Tawny	Crushed
bradley			Articulating	Salmon-colored	Seats	Wood-base	Knot	Two-Tone	Alligator
solitary			Sitting	Mocha-colored	Counter	X-base	Log	Grey	Morocco
lotus			Adornment	Mahogany-color	Buffet	Club	Driftwood	Pastel	Cordovan
prince			White-washed	Straw	Tufted-seat	Wicket	Cedar	Brown	Mocha

Table 2: The result of the model for $\kappa = 5$, before and after Pruning for Furniture category. The top 10 elements in each cluster are shown.

Cluster Depth (C_{depth}): The cluster depth is the depth of the head-node of the cluster in the WordNet sense hierarchy i.e. $C_{depth} = C_{head_{depth}}$.

Cluster Breadth $(C_{breadth})$: The cluster breadth is the vertical span of the the cluster tree in terms of WordNet depth.

$$C_{breadth} = \max(node_{depth} - C_{depth}) \tag{6}$$

where $node \in C$.

Cluster Probability Mass (C_{mass}): The probability of a sense s in catalog d is defined as, $P(s|d) = P(w|d)$ where $s \in \psi_w$ i.e. s is a sense of word w. The cluster probability mass, C_{mass} is based on this.

$$C_{mass} = \sum_{s \in C} P(s|d) \tag{7}$$

Cluster Density $(C_{density})$: The cluster density is defined as:

$$C_{density} = \frac{C_{mass}}{C_{breadth}} \tag{8}$$

Mutual Information $(MI(C_1, C_2))$: Mutual information between any two clusters C_1 and C_2 for a given catalog d measures the amount of common mass between the two clusters. A common word set, Γ_{C_1,C_2}, between C_1 and C_2 is defined as:

$$\Gamma_{C_1,C_2} = \{w : w \in d \wedge (s_1, s_2 \in \psi_w \; s.t. \; (s_1 \in C_1 \wedge s_2 \in C_2))\} \tag{9}$$

where w is a word in catalog d. The Mutual information, $MI(C_1, C_2)$ is defined as :

$$MI(C_1, C_2) = \frac{\sum_{w \in \Gamma_{C_1,C_2}} P(w|d)}{C_{1_{mass}}} \tag{10}$$

To obtain a more sense specific set of clusters with valid attribute labels, the model constraints the above defined cluster properties through 3 model parameters. These parameters are based on the semantic metrics mentioned in section 5.1.. The 3 parameters are as follows:

1. δ: This parameter regulates the depth of discovered cluster C. As observed in section 5.1., the deeper a cluster, the more sense specific it becomes. δ tunes the depth property of a cluster to get suitable product attributes as respective clusters.

2. β: In the WordNet sense-hierarchy, the sense-specificity spreads as one goes down the hierarchy. The model parameter β controls such a spread in the clusters and doesn't let it cross a threshold.

3. μ: The mutual information, $MI(C_1, C_2)$, between 2 clusters C_1 and C_2 gives an estimate of how much one cluster replicates the other. If this replication goes beyond a threshold the smaller cluster should be discarded as it does not contain any independent information of its own. The model parameter μ regulates this threshold.

ALGORITHM 1: Cluster Discovery

Initialize:
Ψ /*obtained via equation 5*/
$\Xi \leftarrow \{\}$/*stores clustered senses*/
$\Omega \leftarrow \{\}$/*set of discovered sense clusters*/
$flag \leftarrow$ false
for *each* $s \in \Psi$, **do**
  for *each* $C_i \in \Omega$, **do**
    if $hypernym(s) = C_{i_{head}}$ **then**
      insert s into cluster C_i
      $flag \leftarrow$ true
    end
    if $hypernym(C_{i_{head}}) = s$ **then**
      $C_{i_{head}} \leftarrow s$
      $flag \leftarrow$ true
    end
  end
  if $flag$ **then**
    insert s into Ξ
    remove s from Ψ
    $flag \leftarrow$ false
  end
  else
    for *each* $s_j \in \Psi$ **do**
      if $hypernym(s) = s_j$ **then**
        create new cluster C_{new}
        $C_{new_{head}} \leftarrow s_j$
        Insert s in C_{new}
        $flag \leftarrow$ true
        break
      end
      else if $hypernym(s_j) = s$ **then**
        create new cluster C_{new}
        $C_{new_{head}} \leftarrow s$
        Insert s_j in C_{new}
        $flag \leftarrow$ true
        break
      end
    end
  end
  if $flag$ **then**
    insert s, s_j into Ξ
    remove s, s_j from Ψ
    insert C_{new} into Ω
  end
end
return Ω

Figure 2 shows depths of all attributes of Toys and Furniture obtained from Wikipedia pages [1] [2]. The attributes obtained from these pages are first converted to their closest morphological noun form. The horizontal lines show the average depth of the categories which is 6.3 for Furniture and 6.7 for Toys. We can also see that the depths are also in the range of 5 and 8. This gives the intuition that the WordNet depths of product attributes are not all that random and have some patterns to them. The model captures this through the parameter δ

[1] `http://en.wikipedia.org/wiki/List_of_furniture_types`

[2] `http://en.wikipedia.org/wiki/List_of_toys`

ALGORITHM 2: Cluster Pruning

Initialize:
$\hat{\Omega} \leftarrow$ [ ] /*array of ranked clusters*/
for *each cluster* $C \in \Omega$ **do**
  $\Lambda \leftarrow$ { }
  if $C_{head_{depth}} \leq \delta \vee C_{breadth} > \beta$ **then**
    $\Lambda \leftarrow DisIntegrate(C)$
    /*function DisIntegrate defined below*/
    remove C from Ω
  end
  for *each cluster* $C_i \in \Lambda$ **do**
    insert C_i into Ω
  end
end
/* Prune for overlapping clusters*/
for *each* $C_i \in \Omega$ **do**
  for *each* $C_j \in \Omega$ **do**
    if $MI(C_i, C_j) > \mu$ **then**
      $C_{smaller} \leftarrow C_j$ **if** $C_{j_{mass}} > C_{i_{mass}}$ **then**
        $C_{smaller} \leftarrow C_i$
      end
      remove $C_{smaller}$ from Ω
    end
  end
end
/* Rank the new clusters*/
$\hat{\Omega} \leftarrow [\Omega]$
for *each* $C_i \in \hat{\Omega}$ **do**
  for *each* $C_j \in \hat{\Omega}$ **do**
    if $C_{i_{density}} \geq C_{j_{density}}$ **then**
      swap $\hat{\Omega}[i]$ and $\hat{\Omega}[j]$
    end
  end
end
return $\hat{\Omega}$
/*function DisIntegrate*/
DisIntegrate(cluster C**):**
$\Lambda \leftarrow \{C\}$
while $\exists\, C_i \in \Lambda \wedge (C_{i_{depth}} \leq \delta \vee C_{i_{breadth}} > \beta)$ **do**
  for *(each* $node_j \in C_i \wedge (C_{i_{depth}} - node_{j_{depth}}) = 1$*)* **do**
    $C_j \leftarrow$ child cluster with head $node_j$ insert C_j into Λ
    remove C_i from Λ
  end
end
return Λ

The parameters, δ, β and μ, used in Algorithm 2 are learned over training set.

Algorithm 2 shows the steps involved in Cluster Pruning phase. Each cluster obtained form phase I is tested on the three parameters δ, β, μ defined above. A cluster which does not satisfy the constraints imposed by any of the three parameters is broken into smaller clusters, where

the smaller cluster are the subtrees one hop down in the sense hierarchy in the cluster. The children of the previous cluster head are the cluster heads of the respective new clusters. When all the clusters present in cluster set, Ω, satisfy the constraints imposed by the model parameters, the model goes on to create a ranked set of clusters $\hat{\Omega}$. In the cluster set $\hat{\Omega}$, the clusters are arranged in the descending order of their cluster density defined in equation 8.
This density is modified in the cases when "hop-holes" are discovered, i.e. when the nearest child to a cluster-head is more than one hop away. In that case, the new density $C'_{density} = \frac{C_{density}}{C_{depth_{avg}} - \delta + hop\_size}$ where $C_{depth_{avg}} = \frac{\sum_{C_i \in \hat{\Omega}} C_{i_{depth}}}{|\hat{\Omega}|}$ and hop_size is the "hop-hole" size.

6. Evaluation Setup

The topics discovered are evaluated by a group of 7 independent domain experts. Each expert-labeler labels every topic discovered and assigns a "valid" or "invalid" label based on whether the topic is a valid attribute of the product. The labelers also label the words present in each valid topic as a "noisy" or "valid" member of the cluster. All the results and parameter-tuning are based on the consensus label of the experts. The consensus label for each data point is obtained via majority voting.

Evaluation Metric. We report number of valid attributes discovered η, and average cluster purity ρ_{avg} of the clusters predicted. The cluster purity of a valid cluster C, if its size is $|C|$ (eg. 100) and has v (say 90) valid words in it as defined above, is $\rho_c = \frac{v}{|C|}$ (0.9). The average cluster purity for top κ clusters is:

$$rho_{avg} = \frac{\sum_{(C \in \ valid)} \rho_c}{\kappa} \tag{11}$$

Data. The model only deals in noun senses to maintain simplicity. All words are converted to their morphologically closest noun word. Eg. "educational" is converted to "education". This does not make the model loose any original word-sense for majority of the words since the model takes all senses of a word into account. Hence all the senses of the new noun-word are taken into consideration reducing the risk of loosing an original word-sense to the minimum. The model tunes its parameters over Furniture and Clothing catalogs. It is tested over Watches and Bedding catalogs and Wikipedia Clay Toy pages.

7. Experiments

7.1. Parameter Tuning.

The parameters δ, β and μ are tuned over two online catalogs: 1) Furniture and 2) Clothing. We take top-40 clusters given by the model i.e. $\kappa = 40$ and count the number of valid clusters. The depth parameter δ is optimised without any β or μ constraints. For this best value of δ the breadth parameter β is tuned without any μ constraint. For these 2 best δ and β the optimal value of μ is tuned. Figure 3 shows the graph for the parameter tuning. The left figure shows that the $\delta = 6$ gives the most number of valid clusters for both catalogs. This result is consistent with the figure 2 where the average WordNet depths for Furniture and Toys category attributes are 6.3 and 6.7 respectively. The center figure in figure 3 shows that for the best δ (6) the optimal β lies in $[6, 8]$. Clothing doesn't show any improvement from constraint β but Furniture gains 2 more valid clusters by imposing β constraint. We take the largest β in $[6, 8]$, $\beta = 8$, as the optimal β to avoid breaking clusters unnecessarily. The right most figure in figure 3 shows the tuning graph for μ for $\delta = 6$ and $\beta = 8$. Furniture doesn't gain anything from μ but Clothing gains 5 more valid clusters by the imposition of μ. Optimal μ lies in the region $[0.6, 0.9]$. We pick $\mu = 0.7$ as optimal as that seems to be optimising for both catalog in the left most figure.

	LDA	Our Model
Bedding	0.50	0.81
Watches	0.35	0.878

Table 4: Average cluster purity ρ_{avg} for $\kappa = 10$

7.2. Test Results

The model is compared with the traditional LDA model and a baseline. The baseline is the number of valid attributes as judged by the experts in top-κ words ranked by the word count in the catalog. However, this baseline would not help the ad-copy creation problem as it just represents a possible label for an attribute set without containing any actual attributes. The word judged to be a valid attribute must be a generic enough word to be a valid label for a cluster of attributes of the product. This baseline is provided solely for comparative study. For the LDA model, each catalog d is divided randomly into N documents with each document getting $\frac{d}{N}$ catalog entries each. The results are reported for best N and optimised parameters of the LDA. Each topic obtained from LDA is a cluster of top 20 most likely words in that topic. We are looking for distinct attributes discovered thus if two topics are about the same attribute then they are counted as one valid topic.

Background. The problem of extracting the product attributes for ad-copy creation involved employees going through the catalog manually and looking for probable attributes which can be formalized in a functional way as described in equation 1. A topic model helped this manual labor by giving a probable set of topics based on the word co-occurrances in the product catalog. One would go through this probable set of topics and extract purer ones by pruning them out. For the purpose of comparison here, we do not prune the topics obtained from LDA and they are reported as "valid" or "noisy" by the experts based on the majority of words being valid or noisy.
We provide an average cluster purity (ρ_{avg}) comparison for valid clusters obtained in the top-10 ($\kappa = 10$) clusters returned by LDA and our model. The clusters reported in table 3 are for $\delta = 6$, $\beta = 8$, $\mu = 0.7$ and $\kappa = \{2, 3, 5, 10, 20, 30, 40\}$. We see that the baseline and LDA are outperformed by our model. The LDA model is only able to find "material" and "brand" attributes for watches catalog and "brand", "size" and "bedding" attributes for bedding catalog. These clusters keep occurring repeatedly in multiple topics discovered by LDA. Table 5 displays the

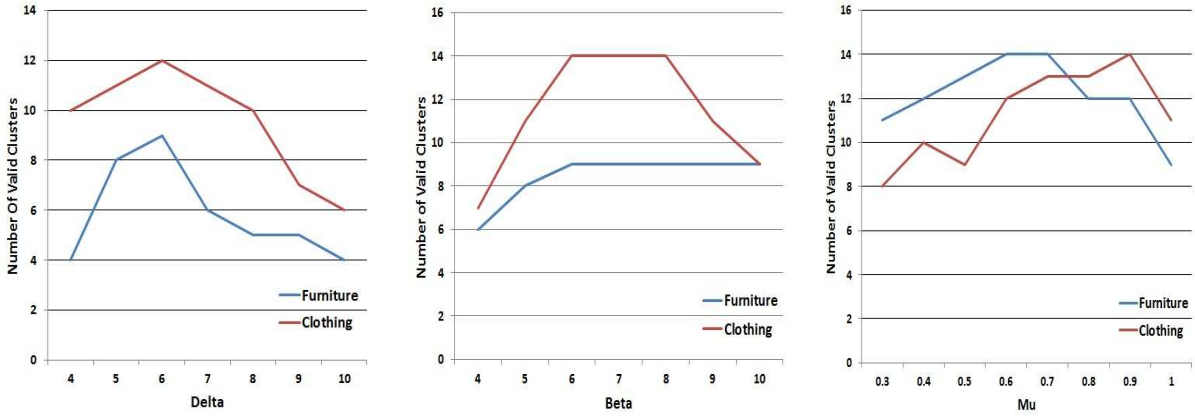


Figure 3: Tuning parameters δ(Delta), β(Beta) and μ(Mu) on Furniture and Clothing catalogs, for $\kappa = 40$.

	$\kappa = 2$	$\kappa = 3$	$\kappa = 5$	$\kappa = 10$	$\kappa = 20$	$\kappa = 30$	$\kappa = 40$
Bedding-LDA	1	1	1	2	2	3	2
Bedding-baseline	1	1	1	1	2	2	2
Bedding-Our Model	1	2	3	5	11	15	19
Watches-LDA	2	2	2	2	2	2	2
Watches-baseline	1	2	2	2	2	2	3
Watches-Our Model	2	3	4	7	8	11	14

Table 3: Number of valid clusters for Bedding and Clothing catalogs for different κ values

timepiece	metal	color	quartz	leather	jewelry	band	Material	Brand
timepiece	metal	color	quartz	leather	jewelry	band	Watch	Roamer
watch	brass	purple	rhinestone	calfskin	bead	rim	Men's	Accutron
timer	steel	red	aventurine	D-KIDS	bling	strap	Women's	Chronotech
clock	bronze	olive	topaz	Grain	pin	Rimmed	Steel	Perpetual
wristwatch	stainless	Brown	agate		band	flat	Stainless	Hush
hunter	gunmetal	salmon	Suede		bracelet	bracelet	Quartz	Crystal-accented
chronograph		blue			clip	weed	Black	Polyurethane
stopwatch		Black			chain	carabiner	Dial	Rotary
alarm		grey			gem		Strap	Luminox
chronometer		yellow			sapphire		Leather	Expansion

Table 5: Valid Clusters discovered for Watches catalog and $\kappa = 10$, the first 7 clusters are discovered by our model and the last 2 are discovered by LDA. The first row in the table is the cluster label.

valid cluster attributes discovered in top-10 clusters given by our model and LDA. We can see that these, attribute clusters are very pure in case of our model. Moreover, these attributes would be very hard to discover by a tagging or a rule based technique unless we know what we are looking for.

8. Discussion and Conclusion

We have presented here an effective mechanism for unsupervised semantics based attribute extraction. The model relies on WordNet semantics and sense-ontology and statistical and unique properties of the SEM dataset. The SEM datsets are a single catalog file containing product entries with each entry effectively a big noun phrase. A word co-occurrance based approach like LDA will not work very well here as shown earlier. The proposed model can also be used as a bootstrapping method for tag based extraction techniques. The valid attribute clusters returned by the model can be used as a seed set for the corresponding attribute set.

Though our model works very well for SEM tasks it has its limitation. It is not a generic model and will fail to extract patterns over a collection of documents. An interesting area of further exploration would be how this model performs for generic topic discovery tasks. This model can be combined with a generative scheme for topic discovery tasks such as LDA in order to make the generative process take into account the semantic properties of words. The current LDA lacks this highly desirable property.

In the present model we assigned same probability of the root word to its child senses. Another way to assign probabilities to sense would be to equally divide the original word's probability among its child sense. This scheme will also take into account the inherent ambiguity of words, i.e. words have with more senses and hence more ambiguous would pass on fewer probability mass to their each child.

9. References

Freitag D. and McCallum A. 1999. *Information extraction using HMMs and shrinkage* In Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction: 31–36.

Xiaofeng Yu, Wai Lam and Bo Chen 2009. *An Algebraic Approach to Rule-Based Information Extraction* In Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM): 325–334.

Reiss F. and Raghavan S. and Krishnamurthy R. and Huaiyu Zhu, and Vaithyanathan S. 2008. *An integrated discriminative probabilistic approach to information extraction* In Proceedings 24th International Conference on Data Engineering (ICDE): 933–942.

Eisenstein J. and Yano T. and Cohen, William W. and Smith, Noah A. Xing, Eric P.; 2011. *Structured Databases of Named Entities from Bayesian Nonparametrics* In Proceedings the EMNLP Workshop on Unsupervised Learning in NLP: 2–12.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates 2005. *Unsupervised named-entity extraction from the web: an experimental study* Artificial Intelligence Journal, 165(1): 91–134.

Fukuda K., Tamura A., Tsunoda T., and Takagi T. 1998. *Toward information extraction: identifying protein names from biological papers* Pac Symp Biocomput, 707–718.

Ellen Riloff and Rosie Jones 1999. *Learning dictionaries for information extraction by multi-level bootstrapping* Proceedings of the 16th national conference on Artificial intelligence (AAAI), 474–479.

K. Bellare, P. Talukdar, G. Kumaran, F. Pereira, M. Liberman, A. McCallum, and M. Dredze 2007. *Lightly-supervised attribute extraction* Proceedings of Machine Learning for Web Search Workshop, NIPS.

Mike Moran and Bill Hunt 2009. *Search Engine Marketing, Inc.: Driving Search Traffic to Your Company's Web Site* IBM Press, Second Edition, ISBN: 978-0-13-606868-6.

George A. Miller 1995. *WordNet: A Lexical Database for English* Communications of the ACM, 38(11): 39–41.

Jay Jiang and David Conrath 1997. *Semantic similarity based on corpus statistics and lexical taxonomy* Proceedings of International Conference on Research in Computational Linguistics, Volume 33.

Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner 1989. *Development and application of a metric on semantic nets* IEEE Transactions on Systems, Man and Cybernetics, 19(1): 17–30.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella 2003. *Kernel methods for relation extraction* The Journal of Machine Learning Research, Volume 3: 1083–1106.

Rayid Ghani, Katharina Probst, Yan Liu, Marko Krema, and Andrew Fano 2006. *Text mining for product attribute extraction* ACM SIGKDD Explorations Newsletter, 8(1).

Istvn T. Nagy and Richrd Farkas 2010. *Person attribute extraction from the textual parts of web pages* Third Web People Search Evaluation Forum (WePS-3), CLEF.

Benjamin Van Durme, Ting Qian, and Lenhart Schubert 2008. *Class-driven attribute extraction* In Proceedings of the 22nd International Conference on Computational Linguistics (COLING): 921–928.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan 2003. *Latent dirichlet allocation* The Journal of Machine Learning Research, Volume 8: 993–1022.

Duangmanee Putthividhya and Junling Hu 2011. *Bootstrapped Named Entity Recognition for Product Attribute Extraction* Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing: 1557–1567.

Joseph Reisinger and Marius Pasca 2009. *Latent Varibale Model for Concept Attribute Attachment* In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: 620–628.

Alexander Maedche, Gnter Neumann, and Steffen Staab 2003. *Bootstrapping an ontology-based information extraction system* Intelligent exploration of the web Physica-Verlag GmbH, Heidelberg, Germany, ISBN:3-7908-1529-2.

David W. Embley, Douglas M. Campbell, Randy D. Smith, and Stephen W. Liddle 1998. *Ontology-based extraction and structuring of information from data-rich unstructured documents* In Proceedings of the 7th international Conference on Information and Knowledge Management (CIKM): 52–59.

Mapping Semantic Relations onto Patterns of Word Use using Corpus Evidence

Patrick Hanks
Research Institute of Information and Language Processing, University of Wolverhampton
Bristol Centre for Linguistics, University of the West of England
Wolverhampton, England
E-mail: patrick.w.hanks@gmail.com

Abstract

This paper presents an empirically well-founded corpus-driven theory of natural language as an analogical system of procedures governed by two interrelated sets of rules: rules for using language normally (idiomatically) and rules for exploiting those norms creatively. The theory is called the Theory of Norms and Exploitations (TNE). Examination of very large quantities of data for word use shows that words in isolation can present unresolvable problems of ambiguity, whereas phraseological patterns, which are not unlike the 'constructions' of construction grammar, are normally each associated with a unique meaning.

Keywords: Theory of Norm and Exploitations, corpus driven, phraseological patterns

1. Introduction

This paper presents an empirically well-founded corpus-driven theory of natural language as an analogical system of procedures governed by two interrelated sets of rules: rules for using language normally (idiomatically) and rules for exploiting those norms creatively. The theory is called the Theory of Norms and Exploitations (TNE). The rules are probabilistic.

Norms are conventions on which members of a speech community mutually rely in order to communicate with one another. To ordinary members of a speech community norms tend to seem basic and obvious, but in fact pre-corpus grammars and dictionaries did not do a very good job of describing them, and even in corpus-based dictionaries confusion is evident, due in part to the absence of a sound theoretical foundation.

Words in isolation can present unresolvable problems of ambiguity and attempting to add 'disambiguation criteria' by speculation ex post facto has proved less than satisfactory (see, for example, Ide and Wilks (2005). On the other hand, empirical examination of very large quantities of data for word use shows that phraseological patterns, which are not unlike the 'constructions' of construction grammar, are normally each associated with a unique meaning and can be used effectively for assignment of meaning to clauses in previously unseen texts.

There are two kinds of 'norm' in TNE. The first type consists of a 'corpus-driven cognitive profile' for every noun that denotes an entity (rather than an event or a state of affairs: nouns of this latter class may be regarded as 'verbs in disguise'). A corpus-driven cognitive profile consists of a sequence of factual statements built around a selection of statistically significant collocates. The set of corpus-driven profiles for the noun *file* are given, showing how the collocates disambiguate the noun.

The second type of norm consists of a predicator (a verb, predicative adjective, or event noun) embedded in a pattern of idiomatic phraseology. A pattern of this kind consists of a valency in which each clause role selects a preferred set of lexical items (nouns and noun phrases) according to their semantic type and/or semantic closeness to a prototypical argument. A procedure that first finds the patterns for a predicator and then attaches an 'implicature' or meaning to the pattern produces more satisfactory results than a procedure that starts with a 'check-list' of meanings for each word and then tries to develop disambiguating procedures.

The second part of the theory concerns the creative exploitation of norms. Every norm has the potential to be exploited creatively by ordinary writers and speakers, as well as poets, novelists, and journalists. Exploitation rules for creative use include metaphors, similes, and other figures of speech, puns, anomalous arguments, and ellipsis.

Both norms and exploitations can be identified by painstaking corpus analysis. However, we must not expect a sharp dividing line between norms and exploitations: some uses of a word are more normal; others are more creative. There is a cline from normal to creative.

2. Related Work

This presentation of a computationally realistic approach to identifying meanings in text has its foundation deep in the history of European structuralism and the work of J. R. Firth (1950, 1957), who argued that "You shall know a word by the company it keeps" and "We must separate from the mush of general goings-on those features of repeated events which appear to be part of a patterned process." Accurate corpus-driven analysis of meaning in text requires a reliable valency grammar—ours is based on Tesnière (1959)—and accurate identification of matters such as clause roles, rank shift, and exponence as outlined in Halliday (1961). Most important of all is the empirical investigation of collocations of John Sinclair (1966, 1984, 1988, 1991, 1998, 2004). In his posthumously published paper (2010) Sinclair argues that phrases rather than words must be regarded as the main meaning-carrying elements of a language. This is a central theme in the work on 'formulaic language' of Wray (2002, 2008), who shows that speakers and writers rely heavily on semi-

preconstructed phrases ('formulas'), rather than building up utterances from basic syntactic principles. This is very much in line with the argument of Carpuat and Wu (2008) that, for machine translation, the aim must be 'PSD' (phrase sense disambiguation) rather than word sense disambiguation.

Corpus-driven analysis of English words and phrases began with the Cobuild dictionary (Sinclair, Hanks, et al., 1987) and has since been elaborated by many, for example Stubbs (2001).

Phraseology is notoriously fuzzy and variable, so some form of computational statistical analysis of the ways in which words normally go together is essential. Such an analysis was first undertaken by Church and Hanks (1989), using pointwise mutual information (PMI) as a statistical measure of word association in text. Since then, many other approaches have been developed, using not only PMI but also other statistical measures of association. Among the most important and user-friendly tool is the Sketch Engine of Kilgarriff et al. (2004), which will be used in the present paper.

In Popescu et al. (2007a, 2007b) a methodology for acquiring sense-discriminative patterns automatically from a corpus is described. Due to an inherent property regarding the sense of the words matching any one of these patterns, called chain clarifying relationships, this methodology can be applied in computational linguistic tasks where the meaning of a phrase plays a major role, for example for sense disambiguation, phrase translation, and textual entailment.

The approach presented here draws on three further components for effective lexical analysis: Preference Semantics (Wilks, 1973), Frame Semantics (Fillmore, 1976, 2006), and Generative Lexicon theory (Pustejovsky, 1995). Recently, Fillmore has argued (Fillmore et al. 2011) that, in addition to a lexicon, linguistics need a 'construction'—an inventory of constructions.

Further theoretical and practical details of the approach to lexical analysis that underlies the presentation here of mapping meanings onto words and phrases will be found in Hanks and Pustejovsky (2005). The preliminary results are publicly available for over 700 verbs of English in an ongoing corpus-pattern analysis research project, the *Pattern Dictionary of English Verbs* (http://nlp.fi.muni.cz/projects/cpa/).

3. Data and Theory

During the 1980s and 1990s some groups of researchers started to build very large collections of texts in machine-readable form, called 'corpora' (singular: 'corpus'). Foremost among these was the British National Corpus (BNC; http://www.natcorp.ox.ac.uk/), a collection of 100 million words (tokens) of more-or-less contemporary English, including 6 million words of spoken English. This became publicly available in 1994. The aim was to build a corpus that would be "balanced and representative".

Although there were many practical problems and indeed theoretical problems (for example, in defining 'representative'), the BNC largely succeeded in its aims, and is now widely used as a resource by lexicographers and linguists alike. As a corpus of 100 million tokens, it is large enough to give researchers good chances of being able to distinguish significant collocations from chance co-occurrences. There are now quite a few large corpora of English, as well as corpora of other languages. This lays the foundations for very fruitful activities in corpus comparison, and in particular of investigating how words are actually used to make meanings (as opposed to the previous activity of speculating about how they might possibly be used). BNC has been superseded, but nevertheless some researchers (including the present writer) prefer to continue to use BNC, despite the fact that it is now about 20 years old, for the sake of comparability of results in long-term research projects. We do not, however, make the mistake of assuming that BNC is equivalent to the English language as a whole. BNC is only a sample, and vulnerable (like all corpora) to the 'failure-to-find' fallacy: the fact that a particular word, construction, idiom, or other linguistic phenomenon is not found in BNC does not mean that it does not exist. Cautious interpretation of results in such circumstances is advisable. This is particularly true of idioms. Quite often, an idiom that may seem very familiar when we consult our intuitions or a linguistics text book (e.g. *kick the bucket* meaning 'die') turns out to be rare or even non-existent in corpus data. This and other phenomena point to the probability that social salience (the frequency of a word or construction) and cognitive salience (its recallability) are independent variable—possibly even in an inverse relationship: what we recall easily is memorable precisely because it is rare. If this is right, introspection is the worst imaginable source of evidence, for it may be that what we think we do (linguistically) and what we actually do are quite different.

4. Revisiting some basic of English Grammar

Corpus linguists aim to undertake empirical studies of how language is actually used. Ideally, these studies would map easily onto the intuition-based speculations of theoretical linguists. All too often, however, the mapping has proved to be difficult or impossible. Faced with a conflict between empirical evidence and theory, some linguists began to attempt revision of received theories, while others preferred to keep the theory intact and throw away the evidence—or, at any rate, those parts of the evidence that do not conform to the theory.

This is the current state of the art. There is tension between received linguistic theory and evidence of usage. This tension cannot be explained away by classifying all uses of words that fail to fit received theories as "performance errors".

In order to understand how words are used to make meanings, it is necessary to examine very large numbers of uses of each word and find the patterns of usage. Words in isolation have only **meaning potential**, but when a word is put into context and used in earnest,

different features of its meaning potential are activated. The rest of this paper is devoted to exploring how this works in the case of a single word, namely ***file***.

We shall not assume a priori that English clauses are directly related to the propositions of predicate logic, nor that every clause has an underlying 'logical form'. Instead, we shall regard these common assumptions as hypotheses, to be confirmed or disconfirmed by the examination of data. The aim in examining corpus data is to discover recurrent patterns of usage and to see whether a meaning (or at least a default interpretation) can be associated with each pattern.

For reasons that we do not need to go into here, the grammatical apparatus best suited for corpus pattern analysis is based on the slot-and-filler grammar of Michael Halliday (1961), adjusted where necessary to account for details of the data. Clause structure and clause roles play a particularly important part in this kind of analysis. The clause roles of slot-and-filler grammar that we shall use for English are:

S – Subject
P – Predicator (the verb and associated words)
– Object (zero, one, or two)
C – Complement (a phrase that is co-referential with the subject or the object)
A – Adverbial (otherwise known as Adjunct). An Adverbial typically consists of a prepositional phrase, but may also consist of a single word such as 'yesterday' or 'home'. A distinction is made between obligatory and optional adverbials.

It should be mentioned that *–ing* forms present peculiar problems of analysis: a filing clerk is not the same as a clerk who is filing.

We shall not assume a priori that English clauses are directly related to the propositions of predicate logic, nor that every clause has an underlying 'logical form'. Instead, we shall regard these common assumptions as hypotheses, to be confirmed or disconfirmed by the examination of data.

The aim in examining data is to discover recurrent patterns of usage and to see whether a meaning (or at least default interpretations) can be associated with each and, if so, what it is.

In this paper, we shall look in some detail at the evidence for uses of word **file**. The analytical apparatus for analysing nouns is quite different from that used for verbs.

4.1 Noun collocates

Figure 1 shows the statistically salient (i.e. socially salient) collocates and arguments in relation to the noun **file**, created by Adam Kilgarriff's Sketch Engine.

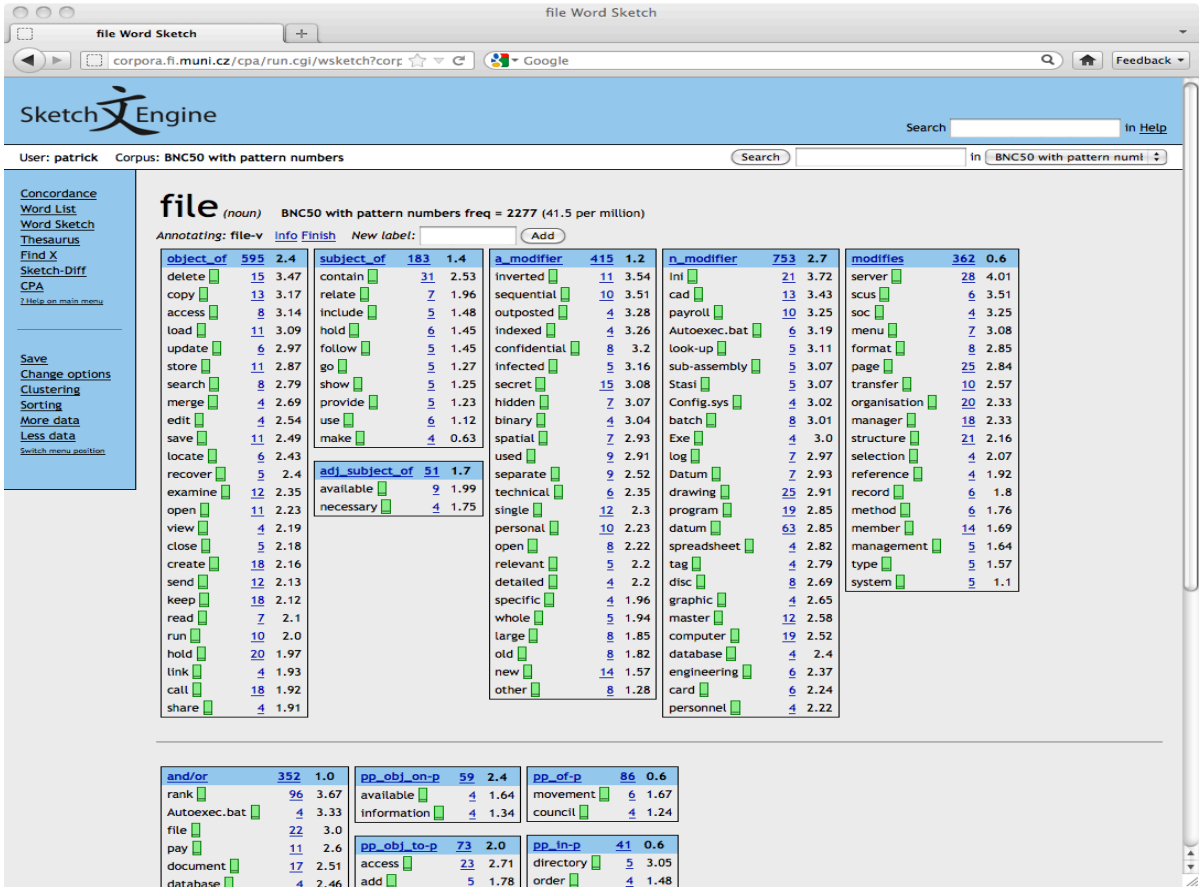


Figure 1. Sketch Engine - Significant collocates of ***file*** as a noun

4.1.1 Noun patterns

Comparing these collocates and the sentences in which they are used yields the following cognitive profiles of the different meaning of the noun **file**. Hypernyms are in boldface; relevant collocates are in italics.

File 1: a file is one of a number of **physical objects**, each containing or intended to contain several paper documents grouped according to an intrinsic criterion such as subject matter, author, date, and/or addressee
Typically, files are held in a *filing cabinet* when they are not in use.
People (typically office workers) *read* or *examine* files in order to find information that is stored in them

File 1.1: a file is an electronic object that is a machine-readable document
Things (actions) that people (= computer users) typically do with files: *create, access, open, load, view, read, search, update, copy, edit, save, close, delete*
People *extract* or *obtain information* from files
If a file contains a computer program, it may be *run* in order to cause the computer to perform some action
A computer file is *stored* or *held* on a *server* or on a *disk* or a *datastick* (*memory stick*)
A program or a computer user may *merge* two or more files
A file may be *sent* to a remote machine or other device
A computer file may be *infected* with *malware*
Computer files have *structure*

File 2: A file can also denote a kind of **tool** with a roughened surface, typically made of tempered steel, used for smoothing or shaping metal or other hard material
A *nail file* is a small tool with a roughened surface, typically an emery board, used for shaping and smoothing one's fingernails.

File 3: A file is a line of people moving one behind the other
People *walk in single file*
The expression *rank and file* is an idiom denoting ordinary human beings in general or the ordinary members of a group (not the leaders)

Figure 2: corpus-driven cognitive profile for *file*, noun

4.2 Verb collocates

The analytical apparatus for deriving verb meanings from a corpus is different from that used for analyzing nouns. The verb is the pivot of the clause, and people construct clauses in order to communicate with one another. Verb meaning is therefore primary. It must be approached by analysing the structures in which the verb is ordinarily used. Such a structure has two components: valency and collocation.

Occasionally (but not always), valency alone is sufficient to distinguish one sense of a verb from another.

1. His lawyer *filed* a lawsuit against Los Angeles city,
2. we all *filed* silently to the Cabinet Room

Thus, with the verb ***file*** the clause-role sequence SPO (as in 1) picks out a different sense of the verb from the sequence SPA (as in 2, where the adverbial particle *to* is the main meaning distinguisher).
However, many meaning differences are not captured by valency analysis. Thus, examples 3 and 4 both have the structure SPO ([NP] file [NP]). However, the meanings are quite different. The meanings are distinguished by the collocates.

3. In 1853 Deacon *filed* his first patent (meaning 'placed on record') .
4. Eleanor was *filing* her nails (meaning 'using a file to shape them').

Filing a document (as in 3) is a quite different action from filing one's nails. We shall now examine in detail the different senses generated when different types of documents are *filed*.

Some verbs, including **file**, generate a hierarchy of increasingly delicate implicatures according to how fine-grained the categorization of the semantic type of the direct object is.

It is clear from the examples given here that there are at least two different verbs in English spelled ***file*** with quite unrelated meanings. As a matter of fact they have different etymologies. Moreover, the first example ('filing a patent') represents the tip of an iceberg, semantically speaking: it is only one of about a dozen different patterns for this verb in the general sense of placing documents on record. The implicatures vary considerably depending on what kind of document is being filed. Moreover, certain inferences can be drawn about the subject of the sentence on the basis of the combination of verb and object, and vice versa. So, for example, if you ***file a lawsuit***, you are assigned the semantic role of being a plaintiff (or the plaintiff's lawyer); if you ***file a tax return***, you have the semantic role of being a taxpayer; if you ***file a story***, you are probably a newspaper reporter; while if you ***file a flight plan***, you do so as the pilot or captain of an aircraft.

In all such cases, filing the document in question not only places it on record but activates some sort of procedure. Other implicatures fall into place, too, just as the scenes-and-frames semantics of writers such as Minsky (1974) predicted they would. Minsky argued that ordinary world knowledge should be represented in relatively large structures called 'frames', which exemplify prototypical cases. Moreover, Minsky's frames "inherit default assumptions that can be displaced when more specific

information is available". And this too can be applied to the analysis of word meaning. The default meaning of ***file*** is that if somebody files something, they place it on record. But this default meaning can readily be displaced (or elaborated) if we know who is filing what.

When a lawyer files a lawsuit, he or she activates a procedure, but a filing clerk filing papers does not activate any procedure.

The common patterns, implicatures, and lexical sets that are actually found for this transitive verb in this group of senses may be summarized as on the next page. Semantic types are in double square brackets. Each semantic type represents a group of words that form a paradigmatic set by virtue of sharing the same hypernym.

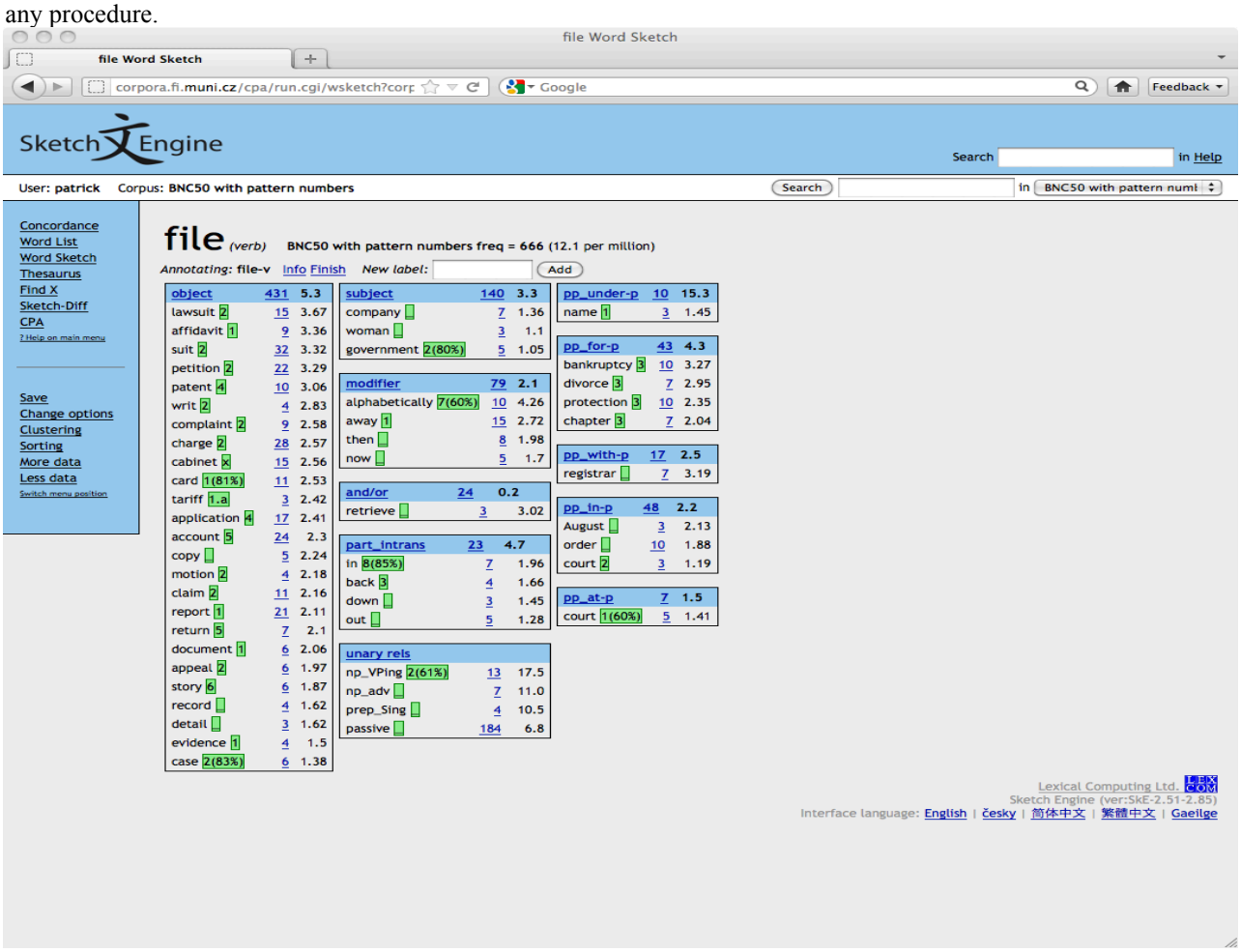


Figure 3. Sketch Engine - Significant collocates of ***file*** as a verb

Basic pattern: **[[Human]] file [[Document]]**
Basic implicature: [[Human]] places [[Document]] officially on record

The following patterns and implicatures account for over 90% of all uses of the verb ***file*** in the corpora I looked at. Other uses, e.g. *people filing into a room*, *people filing notches in bits of wood*, and *people filing their nails* account for less than 10% of the uses of this verb.

1. If verb is 'file' and [[Document]] is [[Lawsuit]]:
A) the role [[Plaintiff]] is assigned to [[Human]]
and
B) 'file' implies activating a court procedure in which [[Plaintiff]] hopes that the court will order compensation to be paid to [[Plaintiff]]
Lexical set [[Lawsuit]] = {*lawsuit, suit, countersuit, writ, claim, counterclaim, action, case, appeal, dispute rectification notice, petition, cross-petition*} ([*against* [Legal Entity]]) ([*for* [Compensation]])

2. If verb is 'file' and [[Document]] is [[Complaint]]:
A) the role [[Complainant]] is assigned to [[Human]]
and
B) 'file' implies activating a procedure which ([[Complainant]] hopes) will result in redress or remedial action (typically, punishment of the person complained against) being ordered by a competent authority
Lexical Set [[Complaint]] = {*complaint, charge, proceedings, lien*} [*against* [Legal Entity]]

3. If verb is 'file' and [[Document]] is [[Evidence]]:
A) 'file' implies making [[Evidence]] available for official use by a court or other authority
and
B) Lexical set [[Evidence]] = {*evidence, information, proof of loss, letter of dissociation, request, patent, brief, affidavit, motion, piece of paper, reply, court papers*}

4. If verb is 'file' and [[Document]] is [[Decision]]:
A) the role [[Judge]] is assigned to [[Human]]
and
B) 'file' implies that the [[Judge]] places his or her [[Decision]] regarding a court case officially on record

Lexical set [[Decision]] = {*decision, opinion, dissenting opinion, court order, order, recommendation*}

5. If verb is 'file' and [[Document]] is [[Tax Return]]:
A) the role [[Taxpayer]] (or [[Accountant]] employed by [[Taxpayer]]) is assigned to [[Human]]
and
B) [[Document]] is a calculation of taxes to be paid by [[Taxpayer]]
and
C) 'file' implies that [[Taxpayer]] acknowledges his or her obligation to pay taxes as calculated in [[Document]]
Lexical set [[Tax Return]] = {*return, taxes*}

6. If verb is 'file' and [[Document]] is [[Patent]]:
A) the role [[Inventor]] is assigned to [[Human]]
and
B) 'file' implies that [[Inventor]] seeks legal protection of the profits from [[Invention]]
Lexical set [[Patent]] = {*patent, patent application*}

7. If verb is 'file' and [[Document]] is [[Application Form]]:
A) the role [[Candidate for Membership]] is assigned to [[Human]]
and
B) 'file' implies that [[Candidate for Membership]] seeks admission to a [[Human Association]]
Lexical set [[Application Form]] = {*form, entry, application*}

8. If verb is 'file' and [[Document]] is [[Nomination]]:
A) the role [[Candidate for Political Office]] is assigned to [[Human]] (by triangulation)
and
B) 'file' implies that [[Candidate for Political Office]] places on record his or her
intention to run for office
Lexical set [[Nomination]] = {*nomination, nomination paper*}

9. If verb is 'file' and [[Document]] is [[Flight Plan]]:
A) the role [[Pilot]] or [[Flight]] is assigned to [[Human]]
and
B) 'file' implies activating a procedure by which official permission to fly the course planned is sought from ground control
Lexical set [[Flight Plan]] = {*flight plan*}

10. If verb is 'file' and [[Document]] is [[Story]]:
A) [[Human]] is newspaper reporter
and
B) [[Story]] is a report of recent events
and
C) 'file' implies sending the text of [[Story]] to the editorial offices of a newspaper for possible publication
Lexical set [[Story]] = {*story, dispatch, column inches, copy*}

11. If verb is 'file' and [[Document]] is [[Paper]]:
A) [[Human]] *may* be assigned the role [[Office Worker]]
and
B) 'file' *may* imply putting [[Paper]]s into a filing cabinet in alphabetical or other order, for storage and possible future retrieval

NOTE: '[[Human]] file [[Paper]]s' is ambiguous. The comparatively low probability of the literal sense is raised dramatically by collocation with 'filing cabinet'

12. If verb is 'file' and [[Document]] is 'report':
'file report' implies no more than that [[Human]] places information on record (with an ambiguous implicature that this may be either in the 'World of Officialdom' frame or the 'Newspaper' frame)
Lexical set [[Report]] = {*report*}

This brings us, finally, to the **default implicature:**

> *If the fine-grained semantic type of [[Document]] is unknown, assume that it is [[Report]] or [[Evidence]], and that 'file' implies putting it officially on record.*

Every verb pattern has a default implicature, and every set of verb patterns has a default implicature at a higher level of generalization.

5. Conclusion

It is a truism that context determines meaning. In this paper, I have tried to put some flesh on this platitude by examining what counts as relevant context, in a linguistic sense. There is another aspect to context, namely context of utterance, which is not analysed here. For context of utterance, readers are advised to turn to the work of Fillmore and to his FrameNet project.

As regards the present work, the meanings of nouns may be summarized by the kind of corpus-driven cognitive profile shown in Fig. 4.2. In the case of verbs, the relationship between context and meaning is found in the relationship between valency (akin to argument structure) and collocation (specifically the sets of noun collocates that are normally used with every verb). Analysis and interpretation are difficult, because in natural language there are no certainties and semantic relationships are not obvious. Sets of collocates may share a semantic type, but as a general rule they are unbounded and set membership is fuzzy. Statistical analysis of large volumes of data computer is therefore an essential first step.

6. Acknoweldgments

This work has been supported in part by the BCROCE project.

7. References

Carpuat, Marina, and Dekai Wu (2008). 'Evaluation of Context-dependent Phrasal Translation Lexicons for Statistical Machine Translation'. Sixth International Conference on Language Resources and Evaluation (LREC-2008), Marrakech.

Church, Kenneth W., Patrick Hanks (1989). 'Word Association Norms, Mutual Information, and Lexicography'. In Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, 26-29 June 1989*. University of British Columbia. Revised version in *Computational Linguistics, 16 (1)*.

Church, Kenneth W., William Gale, Patrick Hanks, Don Hindle, and Rosamund Moon (1994). 'Lexical substitutability'. In B. T. S. Atkins and A. Zampolli (eds.), *Computational Approaches to the Lexicon*. Oxford University Press.

Fillmore, Charles J. (1976). 'Frame semantics and the nature of language'. In *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*.

Fillmore, Charles J. (2006). 'Frame Semantics'. In K. Brown (ed.). *Encyclopedia of Language and Linguistics, 2nd edition*. Elsevier.

Fillmore, C. J., R. Lee-Goldman, and R. Rhodes (2011). 'The FrameNet Constructicon'. In Boas, H.C. and I.A. Sag (eds), *Sign-based Construction Grammar*, Stanford: CSLI Publication.

Firth, J. R. (1950). 'Personality and language in society'. In *The Sociological Review,* xlii.

Firth, J. R. (1957). *Papers in Linguistics 1934-1951*. Oxford University Press.

Halliday, Michael (1961). 'Categories of the theory of grammar'. In *Word 17 (3)*.

Hanks, Patrick (1996). 'Contextual dependency and lexical sets'. In *International Journal of Corpus Linguistics* 1:1.

Hanks, Patrick (2000). 'Immediate context analysis: distinguishing meanings by studying usage'. *ELR Monograph*, University of Birmingham.

Hanks, Patrick (2008). 'Lexical Patterns: from Hornby to Hunston and beyond' (the Hornby Lecture). In *E. Bernal and J. de Cesaris (eds.) Proceedings of the XIII Euralex International Congress*. 9 Sèrie Activitats 20. Barcelona: Universitat Pompeu Fabra, Institut Universitari de Lingüística Aplicada.

Hanks, Patrick (2012). 'How People Use Words to Make Meanings: Semantic types meet Valencies'. In: Alex Boulton & James Thomas (eds), *Input, Process and Product: Developments in Teaching and Language Corpora*. Brno: Masaryk University Press.

Hanks, P., J. Pustejovsky (2005). 'A Pattern Dictionary for Natural Language Processing'. In *Revue Française de Linguistique Appliquée*, 10/2, 63-82.

Ide, N., Y. Wilks (2006). 'Making Sense About Sense'. In Agirre, E. and P. Edmonds (eds), *Word Sense Disambiguation: Algorithms and Applications*, Springer, 47-74.

Kilgarriff, Adam (2004). 'The Sketch Engine' in Euralex Procs. Lorient, France

Minsky, Marvin. (1974). 'A framework for representing knowledge'. MIT AI Laboratory Memo 306.

Popescu, Octavian, Bernardo Magnini (2007a). 'Word Sense Disambiguation Using Sense Discriminative Patterns', in *Proceedings of SCAR, NODALIDA.*

Popescu, Octavian, Sara Tonelli, Emanuele Pianta, (2007b). 'Disambiguation based on Chain Clarifying Relationship Contexts', in *Proceeding of SEMEVAL, ACL*, 2007

Pustejovsky, James. (1995). *The Generative Lexicon*. MIT Press.

Sinclair, John. (1966). 'Beginning the study of lexis'. In C. E. Bazell, J. C. Catford, M. A. K. Halliday, and R. H. Robins (eds.), *In Memory of J. R. Firth*.

Sinclair, John. 1984. 'Naturalness in language'. In J. Aarts and W. Meijs (eds.), *Corpus Linguistics: Recent Developments in the Use of Computer Corpora in English Language Research*. Rodopi.

Sinclair, John (ed.) (1987). *Looking Up: an Account of the COBUILD Project in Lexical Computing*. HarperCollins.

Sinclair, John (1991). *Corpus, Concordance, Collocation*. Oxford University Press.

Sinclair, John (1998). 'The lexical item' in E. Weigand (ed.), *Contrastive Lexical Semantics*. Benjamins.

Sinclair, John (2004). *Trust the Text: Language, Corpus and Discourse*. Routledge.

Sinclair, John (2010). 'Defining the definiendum'. In G.-M. de Schryver (ed.), *A Way with Words: Recent Advances in Lexical Theory and Analysis.* Kampala and Ghent: Menha Publishers.

Sinclair, John, Patrick Hanks(1987). *Collins observed that "You shall know a word by the company it keeps" and "Cobuild English Language Dictionary.* Collins.

Stubbs, Michael (2001). Words and Phrase: Corpus Studies of Lexical Semantics. Blackwell.

Tesnière, Lucien (1959). *Éléments de syntaxe structurale.* Klincksieck.

Wilks, Yorick (1973). 'Preference semantics'. In E. Keenan (ed.), *The Formal Semantics of Natural Language.* Cambridge University Press.

Wray, Alison (2002). *Formulaic Language and the Lexicon.* Cambridge University Press.

Wray, Alison (2008). *Formulaic Language*: *pushing the boundaries*. Oxford University Press.