Improving Domain-specific Entity Recognition with Automatic Term Recognition and Feature Extraction

Ziqi Zhang[a], José Iria[b], Fabio Ciravegna[a]
[a] Department of Computer Science, University of Sheffield, UK
[b] IBM Research, Zurich[1]
E-mail: z.zhang@dcs.shef.ac.uk, jir@zurich.ibm.com, f.ciravegna@dcs.shef.ac.uk

Abstract

Domain specific entity recognition often relies on domain-specific knowledge to improve system performance. However, such knowledge often suffers from limited domain portability and is expensive to build and maintain. Therefore, obtaining it in a generic and unsupervised manner would be a desirable feature for domain-specific entity recognition systems. In this paper, we introduce an approach that exploits domain-specificity of words as a form of domain-knowledge for entity-recognition tasks. Compared to prior work in the field, our approach is generic and completely unsupervised. We empirically show an improvement in entity extraction accuracy when features derived by our unsupervised method are used, with respect to baseline methods that do not employ domain knowledge. We also compared the results against those of existing systems that use manually crafted domain knowledge, and found them to be competitive.

1. Introduction

Entity recognition is the task of identifying and classifying atomic text elements into predefined categories. It often serves as a fundamental step for complex Natural Language Processing (NLP) applications such as information retrieval, question answering, and machine translation. Entity recognition has been applied to domain-specific corpora for recognizing domain-specific entities types such as gene and protein names, and cell types in bioinformatics (Saha et al., 2009; Nenadić et al., 2003), and archaeological artifacts, monument types in archaeology domain (Jeffery et al., 2008; Byrne, 2007). It is generally agreed that domain-specific entities in technical corpora are much harder to recognize and results have been less satisfactory. This is due to the intrinsic complexity of terms in different domains because of multiword expressions, spelling variations, acronyms, ambiguities and so on (Saha et al., 2009; Roberts et al., 2008, Collier et al., 2000).

To address this issue, studies have focused on exploiting domain-specific rules and knowledge resources, such as gazetteers and dictionaries. For example Lin et al. (2004) used domain-specific rules for refining entity boundaries detected by a supervised entity classifier; Seki and Mostafa (2003) used domain-specific rules to identify protein names; Roberts et al. (2008) used terms extracted from the biomedical knowledge resource - UMLS (Lindberg et al., 1993) as a feature in domain-specific entity recognition. Unfortunately, these approaches suffer from limited or no domain portability, since specifically designed rules are limited to the domain-in-question and domain-specific knowledge resources are often not available or cannot be exploited in the same way in different domains. Furthermore, they are often difficult to build and maintain (Kazama and Torisawa, 2008).

In this paper, we introduce a novel approach to domain-specific entity recognition, which avoids the effort in gathering, analyzing, building and maintaining domain-specific knowledge resources and rules, and empirically show that the approach consistently improves entity recognition accuracy across corpora in different domains. To achieve this, we hypothesize an association between the domain-specific entities we want to recognize, and domain-specificity of the words in the corpus. We measure domain-specificity of words by Automatic Term Recognition techniques (ATR, Korkontzelos et al., 2008; Zhang et al., 2008), and use it as a feature in a supervised entity classifier. As in previous related work, the intuition is that domain-specific entities tend to be composed of or co-occurring with domain-specific words within a context window. For example, research on biomedical entity recognition has shown that biomedical entities are often modified by certain so-called "trigger" words that are domain specific (Spasić et al., 2003; Fleischman and Hovy, 2002; Jiang et al., 2006). However, contrary to previous approaches, we use a completely unsupervised method to identify the domain-specific words. We validate the proposed approach on four different corpora in two different domains, and show improved performance over baseline systems that do not employ domain-specific features. On the three corpora for which state-of-the-art results are publicly available, we obtain competitive results to those reported by systems that make extensive use of manually-built, domain-specific knowledge resources and rules.

The rest of this paper is organized as follows: section 2 introduces our method. Section 3 compares our method with related work. Section 4 covers experiments and results. Section 5 is discussion, and we conclude our work in section 6.

2. Methodology

Our approach is based on the hypothesis that domain-specific named entities tend to be either composed of domain-specific words, or indicated by

[1] This author carried out this work while being a former member of the Department of Computer Science, University of Sheffield

domain-specific words within a context window. To exploit this insight, the simplest approach is to apply ATR to extract domain-specific terms and use them as lookup-lists or gazetteers in the context of a classification task (classifying mentions of entities in the corpus), as done by Jiang et al. (2006), Nenadić et al. (2003) and Saha et al. (2009). However, the main disadvantage of this class of approaches is that they rely on a hard decision about domain-specificity of terms to happen before the entity extraction task itself. As noted in Frantzi et al. (1998), it is often difficult to distinguish domain-specific terms from non-terms. Thus, terms incorrectly recognized and accepted into the gazetteer introduce noise when learning the entity classifiers. To avoid this, we instead apply ATR algorithms to compute a *domain-specificity* score for each word, and use the scores to produce a ranked list of terms. Then, we can use information derived from a term's rank as a *feature* for the learning algorithm, as explained below. This effectively merges the term extraction step with the entity extraction step, relying on the robustness of the learning algorithm to identify which terms are discriminative by taking account the training data, which is otherwise unavailable in a separate term extraction step.

The ranked list of terms obtained from the output of the ATR module, if used directly, i.e., treating each rank as a binary feature, would induce a high-dimensional space, which is known to limit the generalization performance and worsen running time requirements of learning algorithms (Fodor, 2002; Saha et al., 2009). For that reason, we adopt a feature extraction method that attempts to transform the original high dimensional feature space into a lower dimensional space preserving or improving the discriminative ability of the classifier. The method can also be interpreted as a clustering method that partitions the features into several clusters. In the following, we describe the above in details.

2.1 ATR algorithms

We choose two ATR algorithms, which are term frequency *tf* (Dagan and Church, 1994), and term frequency versus inverse document frequency *tf-idf*, as used in an ATR setting (Zhang et al., 2008). We choose these two algorithms because they are easier to implement than other alternatives such as *C-Value* (Frantzi et al., 1998), have long been accepted as state-of-the-art ATR algorithms, and have proved sufficient in our experiments to verify the validity of our approach. Thus, our purpose is not to find the maximum performance attainable by the choice of ATR algorithms; rather, we aim to demonstrate that they can be used to improve domain-specific entity recognition. Traditionally, domain-specific ATR techniques can be applied at both document-level and corpus level. For the purposes of feature generation, we obtain statistics from each document instead of from the whole corpus.

Thus for a given word w_i in a document d_j, we compute *tf* of $w_{i,j}$ as the frequency of w_i in d_j, denoted by $tf_{i,j}$, as follows:

$$tf_{ij} = \frac{w_{i,j}}{\sum_k w_{k,j}} \qquad (1)$$

where the denominator is the sum of all words found in document d_j. And we compute *tf-idf* of a word w_i in document d_j, denoted by $tfidf_{i,j}$ using:

$$tfidf_{i,j} = tf_{i,j} \times idf_{i,j} \qquad (2)$$

where $tf_{i,j}$ is measured using equation 1 and $idf_{i,j}$ is computed as:

$$idf_{i,j} = \log \frac{|D|}{|\{d \in D : w_i \in d\}|} \qquad (3)$$

in which $|D|$ is the total number of documents in the corpus, $|\{d \epsilon D : w_i \epsilon d\}|$ is the number of documents in which word w_i is found.

We use each algorithm to compute a double score of domain-specificity ds_w of the target word w, then rank these scores and use the rank as input to the feature extraction method described in the next section, to generate the binary features for the entity extraction classifier.

2.2 Feature extraction

To reduce the dimensionality, we apply feature extraction to transform the original *n*-dimensional feature space into a lower-dimensional space, where *n* is the number of terms output by the ATR algorithm (typically in the order of hundreds of thousands). Following Magalhães and Rüger (2007), we define a feature extraction function *f* to reduce the number of dimensions of a sparse space with dimensionality *n* into a much lower dimension $k << n$, formalized as:

$$f(w_1,\ldots,w_n) = \begin{bmatrix} f_1(w_1,\ldots,w_n) \\ \ldots \\ f_k(w_1,\ldots,w_n) \end{bmatrix} \qquad (4)$$

To achieve this, we rank the list of terms outputted by the ATR module in decreasing order of the ds_w score. Let $r_{ds}(w)$ denote the rank of term *w* and *R* denote the total number of elements in the list. For our purposes, we define the *k* feature extraction functions of equation 4 as:

$$f_i(w_1,\ldots w_n) = \begin{cases} 1, \exists\, j: \frac{R}{k} \times (i-1) < r_{ds}(w_j) \le \frac{R}{k} \times i \\ 0, otherwise \end{cases} \qquad (5)$$

where $i=1,..,k$. This can be interpreted as dividing the ranked list evenly into *k* partitions, and setting an indicator function for a given partition *i* if there is a term whose rank falls into that partition. This technique can also be viewed as a feature clustering method, such that we define number of clusters as *k*, and we use information derived from the domain-specificity of words and partitions of the ranking as a simple metric to compute distance between terms.

3. Related Work

Our work is most related to the application of ATR techniques and feature extraction techniques in entity

recognition. Compared to state-of-the-art domain-specific entity recognition systems that rely on domain-specific features and resources, our method generates domain-specific knowledge in an unsupervised way and integrates it into machine learning algorithms as a useful feature. In addition, compared to other feature extraction techniques, our approach to feature extraction requires less computation due to its simplicity and is thus suitable for large scale applications.

ATR is a well-developed research field that deals with extraction of technical terms from domain-specific language-corpora (Korkontzelos et al., 2008). The major difference between ATR and entity recognition is that ATR does not aim to classify terms, but measures how likely a term is specific to the domain in question. The advantage of ATR is that it generates domain-knowledge – in the form of domain specificity of terms – without relying on the availability of domain-specific external re-sources.

Typically, statistical measures, such as term frequency (Dagan and Church, 1994), term frequency versus inverse document frequency (Jones, 1972) and mutual information (Damerau, 1990), are employed to measure domain specificity of a term. Prior work on entity recognition has employed some of these techniques to improve performance. Many researchers such as Fleischman and Hovy (2002), Saha et al (2009), Jiang et al (2006) and Spasić et al. (2003) make use of frequency statistics observed in the corpus to extract so-called "trigger" words, and use them as a feature. For example, Spasić et al. (2003) firstly apply C/NC-Value (Frantzi et al., 1998) to recognize domain-specific terms from a biomedical corpus; next, they extract domain-specific verbs using their frequency in the corpus and co-occurrence frequency with recognized terms; then they used a classifier to classify these terms into several classes, and finally the classes of terms that are selected as arguments for the considered verbs are induced. The argument for using trigger words is that domain-specific entities are often modified by certain words than by chance. These methods focus on extracting features from ATR for contextual words of target instance and are usually selective about the words either by word classes or setting an arbitrary threshold; however, our approach applies the features to both context and target words because the intuition is that domain-specificity of words composing entities are as important as contextual words. Also, as described before, due to the difficulty of identifying boundaries between terms and non-terms, our method does not select features arbitrarily; rather we rely on the nature of entity classifiers to select appropriate features by learning from examples.

The most similar work to ours is Finkel et al. (2004), who observe that in the task of recognizing biomedical entities, many entities in the domain-specific gazetteers are ambiguous and cause noise to the entity classifier. To solve the problem they make use of frequency of words, by associating a frequency category according to the observed in a general purpose corpus – the British National Corpus, and use it as a feature in the entity classifier. Unfortunately details of their method for categorizing word frequency are unknown and thus it is difficult to compare their method with ours. One advantage of our method is that we do not rely on additional resource other than the corpora of interest. Nenadić et al (2003) argue that domain specific terms represent important concepts in a domain, that they are "semantic indicators used in scientific discourse", and thus could be useful features to entity recognition. In their experiment, an ATR algorithm is first run to extract domain specific terms, which are then used as a feature to an entity recognition classifier. Again because ATR algorithm can be erroneous, incorrectly extracted terms may become noisy features. In contrast, our method does not require such a separate processing stage, but naturally integrates the features extracted from ATR techniques into the entity classifier.

Da Silva et al. (2004) hypothesized that NEs are often Muti-Word Units (MWUs), and proposed using mutual information measures and frequency of words to identify n-grams (where n>1) from corpus that are potential entities. Later Downey et al. (2007) extended this idea and applied similar method using the Web data. However, these methods do not attempt to classify entities to pre-defined categories.

Concerning feature extraction, it addresses the issue that from a computational learning viewpoint, an increase in feature dimensions leads to data sparsity and poses computational challenges, and what's more, does not always improve performance (Fodor, 2002; Saha et al., 2009). Feature extraction transforms high-dimensional features as input to reduced low-dimensional features. In the studies of entity recognition, feature clustering applied to words has been proposed as a common way of feature extraction. Miller et al. (2004) applied unsupervised clustering technique to derive hierarchical clusters of words from a large unlabelled corpus; later, using cluster membership as features in a supervised entity recognition task significantly improved system performance. Kazama and Torisawa (2008) argue that in entity recognition tasks, a gazetteer is useful if it returns consistent labels even if those are not the predefined entity categories. To exploit this feature they cluster candidate MWUs and use the clusters as sources of gazetteers, which resulted in better performance than their baseline system. Their method is essentially feature extraction through word clustering. Saha et al. (2009) used several techniques to reduce feature dimensionality. First, they proposed to measure the importance of context of a target word using corpus statistics, and only keep the context words whose importance is above a threshold. Second they also used word clustering and used the derived clusters as features. Our method of feature extraction can be considered similar as a word clustering technique. However, we take a very different way of measuring word similarity and clustering; that we consider domain-specificity of words as similarity input to a clustering algorithm, which in our case, is achieved through a simple function based on

ranking.

4. Experiments

In this section, we describe our experimental setting. Our purpose is to prove that features extracted from domain-specificity of words, by using our methodology described in section 2, can be used to improve domain-specific entity recognition. Therefore, we have selected four corpora from two scientific domains, as listed in Table 1.

Corpus name	Domain	Number
archaeo	Archaeology	100 full length articles
Genia JNLPBA04	Biomedical	2400 abstracts
Yapex	Biomedical	200 abstracts
Bio1.1	Biomedical	100 abstracts

Table 1. Corpora selected for experiments and their statistics

We select three corpora from the biomedical domain because of its public availability and popularity in the studies of entity recognition. These include the adapted Genia corpus (including training and testing) for the JNLPBA'04 shared task, which includes 2400 abstracts selected from National Library of Medicine's MEDLINE database. The corpus has been annotated for protein (approx. 34k annotations), cell type (approx. 85k), cell line (approx. 4k), DNA (approx. 10k) and RNA (approx. 1k) entities. The Yapex[2] corpus contains 200 MEDLINE abstracts, 53 of which overlap with Genia but are re-annotated by different annotators. The corpus contains approximately 3700 annotations of protein names. The Bio1.1 [3] corpus contains 100 MEDLINE abstracts, including annotations[4] of protein names (approx. 2k), DNA (approx. 350), and SOURCE (approx. 800), which include 7 sub-types such as "cell type", "cell line", "virus" and so on. In our study we treat them as a single entity type. For the domain of archaeology, we select a subset of corpus used in Jeffery et al. (2009), which contains 100 full length articles archived by the Arts and Humanities Data Service (AHDS[5]). The corpus is annotated for three entity types: archaeological temporal terms (TEM, approx. 4k), such as "Bronze Age", "Medieval", and "1089AD"; location of interest (LOC, approx. 2.5k), which is UK-specific and often refers to place names of findings and events; subject of interest (SUB, approx. 11k), which is highly heterogeneous, containing terms used in various domains, such as architecture, warfare, maritime, education and so on. Each corpus is split into five equal parts for five-fold cross-validation experiments.

Firstly, we show that our proposed method of exploiting ATR and feature extraction techniques improves the baseline system that does not employ any domain-specific knowledge on all four corpora. Next, for the three biomedical corpora on which prior systems have been tested on, we compare our system performance against state-of-the-art systems that make use of various domain-knowledge to show that, by only adding the features extracted from ATR techniques using our method, our system achieves competitive results. For this purpose, we use domain-independent features in our baseline system, and follow the same evaluation methodology (*Precision*, *Recall* and *F-measure (F1)* with exact match) introduced in the JNLPBA04 shared task on bio-entity recognition. The baseline features we tested consist of

- Words within a context window of 5 tokens
- Word lemma
- Word stem
- Word orthographic type
- Token kind (e.g., digit in words)
- Word shape (e.g., capitalized letters replaced by 'A', small characters replaced by 'a') of a word
- Word part-of-speech (POS) – obtained from a generic POS tagger from the OpenNLP[6] tools

In addition, we vary the value of *k* in formula 5 from 0 to 40, with unit increment of 5. We developed our system using the Runes [7] data representation framework for processing the text, a collection of information extraction modules from T-rex [8], and the machine learning framework Aleph[9] for the learning algorithm. We selected a SVM classifier due to its robustness to noisy features and wide availability. Figure 1.1 to 1.4 compares baseline system performance (in *F-measure*, *y*-axis) against baseline plus ATR extracted features under different *k* values (*x*-axis). Table 2.1 to 2.4 compares the best results of our system with results reported in the literature in the domain of biomedical science.

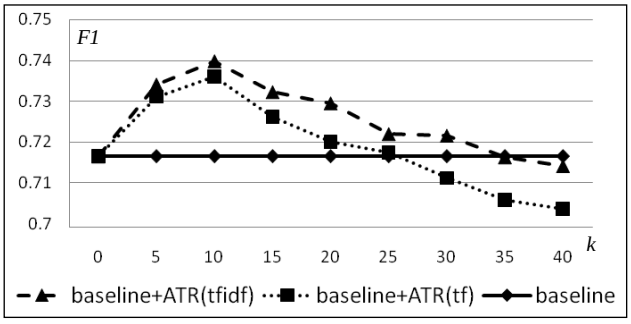

Figure 1.1. Archaeo corpus

[2] http://www.sics.se/humle/projects/prothalt/
[3] http://compbio.uchsc.edu/corpora/obtaining.shtml
[4] We ignored annotations of type 'RNA' as it has less than 40 annotations only
[5] http://ahds.ac.uk/
[6] http://opennlp.sourceforge.net/
[7] http://runes.sourceforge.net/
[8] http://t-rex.sourceforge.net/
[9] http://aleph-ml.sourceforge.net/

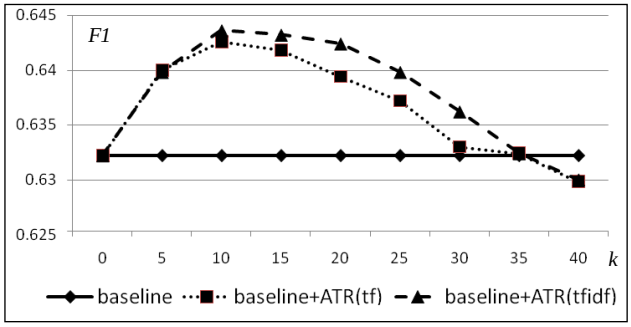

Figure 1.2 Genia JNLPBA04 corpus

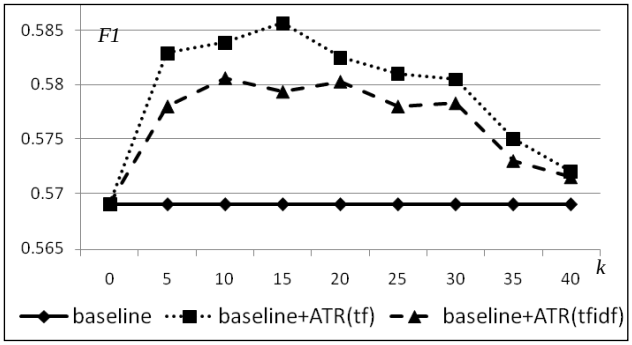

Figure 1.3. Yapex corpus.

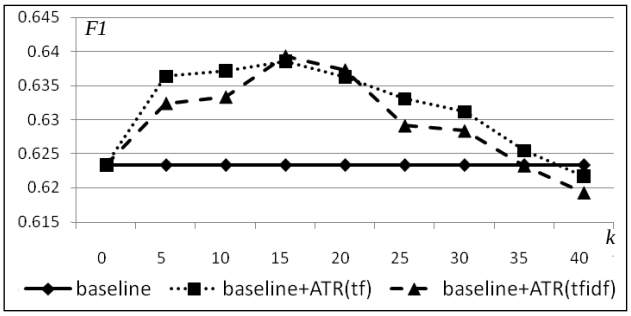

Figure 1.4. Bio1.1 corpus.

	Domain knowledge	LOC	SUB	TEM
Our baseline	None	67.1	65.8	82
Our baseline + ATR feature (best)	None	*69.8 (tfidf, k=10)*	*67.8 (tf, k=15)*	*85.1 (tfidf, k=10)*
Zhang & Iria (2009), baseline	None	68.4	65.7	81.8
Zhang & Iria (2009), baseline + domain specific gazetteer	Manually created *domain gazetteers*	70.5	67.5	82.5

Table 2.1. Performance figures on the archaeo corpus.

System	Domain knowledge	Fm
Our system - baseline	None	63.22
Our system – baseline + ATR feature (best)	None (ATR=tf-idf, *k*=10)	*64.4*
Saha et al. (2009) baseline	POS (Genia POS tagger[10]), prefix and suffix, special characters	64.82
Saha et al. (2009) final	(same as above) + trigger words	65.79
Song et al. (2004) baseline	POS (Genia POS tagger), phrase, prefix & suffix, *word dictionary*	63.85
Song et al. (2004) final	POS (Genia POS tagger), phrase, prefix & suffix, *word dictionary*, Virtual Sample	66.26
Zhou & Su (2004) baseline	POS (Genia POS tagger)	64.1
Zhou & Su (2004) final	Resolution of name alias, cascaded NEs, abbreviations; dictionary; POS (Genia POS tagger)	72.55

Table 2.2. Performance figures on the Genia JNLPBA04 corpus.

System	Domain knowledge	Fm
Our system - baseline	None	56.9
Our system – baseline+ ATR feature (best)	None (ATR=tf, *k*=15)	*58.6*
Chang et al. (2004)	Domain-specific patterns; special character	57.6

Table 2.3. Performance figures on the Yapex corpus

System	Domain knowledge	Fm
Our system – baseline	None	62.3
Our system – baseline+ ATR feature (best)	None (ATR=tfidf, *k*=15)	*63.7*
Collier et al. (2000)	Domain specific letters	63.9[11]

Table 2.4. Performance figures on the Bio1.1 corpus

[10] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

[11] For Bio1.1, the reported average F-measure including protein, RNA, DNA and SOURCE by Collier et al (2000) is 72.8. However, since we ignore the RNA class, thus we take the average of F-measures of protein, DNA and SOURCE as the F-measure of all entity types for this corpus.

5. Discussion

As shown in Figure 1.1 to 1.4, the ATR features extracted by our proposed method improve our baseline system by approximately 1.1 ~ 2 percent in *F-measure* on all corpora; which indicates that the feature exploited from ATR technique is useful to domain-specific entity recognition, and that it is generic and applicable across different domains and on different corpora. Comparing performances at per-entity-type basis, we observe improvements on all entity types. The results also show the optimum *k* value resides in the range 5 ~ 20; and as *k* increases, system performance degrades gradually until below baseline performance, indicating that the values of the extracted ATR features become sparse and less useful to the learning algorithm.

It is unclear which ATR algorithm outperforms the other, possibly indicating that the accuracy of ATR algorithm is not critical, and as long as the algorithm returns consistent ranking of domain-specificity, the entity classifier is able to learn useful patterns from examples. This is an important feature of our approach compared to other approaches that usually make arbitrary thresholds of domain-specificity (e.g., frequency) for selecting useful terms as features.

Compared with state-of-the-art domain-specific entity recognition systems, our system is competitive since it does not rely on any manually created domain-specific knowledge, but produces comparable results. In many occasions, our methods have produced better results than those systems that make use of manually created knowledge. Particularly, our method even outperforms the system that uses manually created gazetteers on the archaeo corpus. (Table 2.1). For the Genia corpus, we compare our system against Saha et al. (2009) and those reported in the JNLPBA04 workshop on the shared task on bio-entity recognition. As shown in Table 2.2, all systems except ours, make use of domain-specific knowledge such as dictionaries, acronym lookup-lists, specially trained POS tagger, word prefix and suffix, and special characters. Although deriving the features of word prefix and suffix and special characters and letters do not rely on domain-specific knowledge, as noted by Collier et al. (2000) and Saha et al. (2009) themselves, these are domain-specific features that are particularly useful in the biomedical domain. We observe that these domain specific knowledge plays an important role in these systems, such that when some of them are unavailable (indicated with the keyword "baseline"), the performance drops by 1 ~ 8 percent in F-measure (row 3 v.s. 4; row 5 v.s. 6; row 7 v.s. 8). Contrary to these systems, our system does not make use of any domain-specific knowledge. Our baseline system produces an *F-measure* of 63.22, which is lower than all other systems. After adding the extracted ATR features, our system reaches 64.4, which outperforms the baseline systems of Zhou et al. (2004), Song et al. (2004) who employ dictionary look-up feature; and is very close to that of Saha et al (2009). Note that these baseline systems still make use of domain specific features and domain-specific POS tagger. According to Saha et al. (2009), adding POS feature generated by Genia Tagger increased their system performance by 1.7 in *F-measure*.

On the Yapex corpus, we compare our system with Chang et al. (2004), which is also an SVM-based entity classifier. Our baseline performance is 56.9 in F-measure, which is 0.7 lower. After adding the extracted ATR features, our system outperforms Chang's system that makes extensive use of domain-specific rules for improved results. And on the Bio1.1 corpus, our system with extracted ATR feature performs almost as well as Collier et al (2000), who used an HMM model and some domain-specific features.

As described before, our approach to feature extraction from the output of ATR can be viewed as using a simple clustering function, defined by formula 5, to cluster domain-specificity of words. The empirically verified performance improvement indicates that such a simple feature space transformation is effective. Moreover, compared to other clustering algorithms applied to word features, our method is much simpler and faster, and is therefore more suitable for large-scale processing.

6. Conclusion

In this paper, we proposed a novel approach to exploit features derived from ATR techniques to improve domain-specific entity recognition. To do so, we use domain-specificity of words measured by ATR algorithms as binary features to a supervised entity classifier.

Since these features induce a high-dimensional space, and are thus too sparse to be useful for learning purposes, we apply a fast feature extraction method to reduce the dimensionality of the feature space. Using these features in our baseline entity classifier improved system performance in the domain-specific entity recognition tasks on four different corpora in two different domains. Compared to state-of-the-art systems in the field, our proposed approach obtained competitive results. Although our system does not always outperform the state-of-the-art systems, the real value of our approach is that it is a generic, unsupervised method for automatically generating domain-specific knowledge and naturally integrating it in the entity classifier, which empirically proves useful across domains. By adding domain-specific features and knowledge resources we expect the system performance to be improved further. We will investigate this as future work.

One limitation of our approach is the requirement for arbitrarily defining the value of *k*. In the future, we will research on automatically deriving the optimum value. Finally, our system proved successful in using domain-specificity measured at word level. Naturally, ATR can be applied to identify multi-word terms. Our expectation is that domain-specificity measured at multi-word level may produce better discriminative features than single-word terms, and will research in this direction in the future.

7. Acknowledgements

This work is funded by the Archaeotools project that is carried out by Archaeology Data Service, University of York, UK and the Organisation, Information and Knowledge Group (OAK) of the Department of Computer Science, University of Sheffield, UK.

8. References

Byrne, K. (2007). Nested Named Entity Recognition in Historical Archive Text. In *Proceedings of International Conference on Semantic Computing*.

Chang, J., Schütze, J., Altman, R. (2004). GAPSCORE: finding gene and protein names one word at a time. In *Bioinformatics 2004*, pp. 216-225.

Collier, N., Nobata, C., Tsujii, J. (2000). Extracting the Names of Genes and Gene Products with a Hidden Markov Model. In *Proceedings of COLING'00*, pp.201-207

Da Silva, J., Kozareva, Z., Noncheva, V., Lopes, G. (2004). Extracting named entities. A statistical approach. *In Proceedings of TALN*

Dagan, I., Church, K. (1994). Termight:Identifying and translating technical terminology. In *Proceedings of the 4th conference on Applied Natural Language Processing*, pp. 34-40

Damerau, F. (1990). Evaluating computer-generated domain-oriented vocabularies. In *Information processing and management: an International journal*, 26(6), pp. 791-801

Downey, D., Broadhead, M., Etzioni, O. (2007). Locating complex named entities in Web text. *In Proceedings of IJCAI'07*.

Finkel, J., Dingare, S., Nguyen, H., Nissim, M., Manning, C., Sinclair, G. (2004). Exploiting context for biomedical entity recognition: from syntax to the web. In *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*

Fleischman, M., Hovy, E. (2002). Fine grained classification of named entities. In *Proceedings of COLING'02*, pp. 1-7.

Fodor, I. (2002). A survey of dimension reduction techniques. Technical report, Lawrence Livermore Nat Laboratory, Center for Applied Scientific Computing.

Frantz, K., Ananiadou, S., Tsujii, J. (1998). Automatic Recognition of Multi-Word Terms: the C-value/NC-value method. In *International Journal on Digital Libraries* Vol. 3, No. 2, pp.115—130.

Jiang, W., Guan, Y., Wang, X. (2006). Improving feature extraction in named entity recognition based on maximum entropy model. In *Proceedings of the 5th International conference on machine learning and cybernetics*.

Jeffrey, S., Richards, J., Ciravegna, F., Chapman, S., Zhang, Z. (2009). The Archaeotools project: Faceted Classification and Natural Language Processing in an Archaeological Context. *Special Theme Issues of the Philosophical Transactions of the Royal Society A,"Crossing Boundaries: Computational Science, E-Science and Global E-Infrastructures"*

Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation 28 (1): 11–21*.

Kazama, J., Torisawa, K. (2008). Inducing Gazetteers for Named Entity Recognition by Large-scale Clustering of Dependency Relations. In *Proceedings of ACL-2008: HLT*, 407-415.

Korkontzelos, I., Klapaftis, I., Manandhar, S. (2008). Reviewing and Evaluating Automatic Term Recognition Techniques. In *Proceedings of the 6th International Conference on Natural Language Processing, GoTAL2008*

Lindberg, D., Humphreys, B., McCray, A. (1993). The Unified Medical Language System. In *Methods of Information in Medicine*, 32(4):281–291

Lin, Y., Tsai, T., Chou, W., Wu, K., Sung, T., Hsu, W. (2004). A Maximum Entropy Approach to Biomedical Named Entity Recognition. In *Proceedings of BIOKDD'04, 4th Workshop on data mining in bioinformatics* (with SIGKDD'04)

Magalhães, J., Rüger, S. (2007) Information-theoretic semantic multimedia indexing. In *Proceedings of CIVR'07*.

Miller, S., Guinness, J., Zamanian, A. (2004). Name tagging with word clusters and discriminative training. In *Proceedings of HLT'04*.

Nenadić, G., Rice, S., Spasić, I., Ananiadou, S., Stapley, B. (2003). Selecting text features for gene name classification: from documents to terms. In *Proceedings of ACL workshop on NLP in biomedicine*, 13, pp. 121-128.

Roberts, A., Gaizauskas, R., Hepple, M., and Guo, Y. (2008). Combining Terminology Resources and Statistical Methods for Entity Recognition: an Evaluation. In *Proceedings of LREC2008*.

Saha, S., Sarkar, S. & Mitra, P. (2009). Feature selection techniques for maximum entropy based biomedical named entity recognition, Journal of biomedical informatics, 42(5), pp.905-911

Seki, K., Mostafa, J. (2003). A probabilistic model for identifying protein names and their name boundaries. In *Proceedings of the IEEE Computer Society Conference on Bioinformatics*

Song, Y., Kim, E., Lee, G., Yi, B. (2004). POSBIOTM-NER in the shared task of BioNLP/NLPBA 2004. In *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*

Spasić, I., Nenadić, G., Annaniadou, S. (2003). Using domain-specific verbs for term classification. In *Proceedings of ACL Workshop on NLP in biomedicine*, 13, pp. 17-24.

Zhang, Z., Iria, J. (2009). A novel approach to automatic gazetteer generation using Wikipedia. *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources. pp.1-9*

Zhang, Z., Iria, J., Brewster, C., Ciravegna, F. (2008). *A comparative evaluation of term recognition algorithms*. In *Proceedings of LREC'08*

Zhou, G., Su, J. (2004). Exploring deep knowledge resources in biomedical name recognition. In *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*