Building a Bilingual Representation of the Roget Thesaurus for French to English Machine Translation

Violaine PRINCE, Jacques CHAUCHE

University Montpellier 2 / LIRMM-CNRS
161 Ada Street, 34392 Montpellier cedex 5 FRANCE
prince@lirmm.fr, chauche@lirmm.fr

Abstract

This paper describes a solution to lexical transfer as a trade-off between a dictionary and an ontology. It shows its association to a translation tool based on morpho-syntactical parsing of the source language. It is based on the English Roget Thesaurus and its equivalent, the French Larousse Thesaurus, in a computational framework. Both thesaurii are transformed into vector spaces, and all monolingual entries are represented as vectors, with 1000 components for English and 873 for French. The indexing concepts of the respective thesaurii are the generation families of the vector spaces. A bilingual data structure transforms French entries into vectors in the English space, by using their equivalencies representations. Word sense disambiguation consists in choosing the appropriate vector among these 'bilingual' vectors, by computing the contextualized vector of a given word in its source sentence, wading it in the English vector space, and computing the closest distance to the different entries in the bilingual data structure beginning with the same source string (i.e. French word). The process has been experimented on a $20,000$ words extract of a French novel, *Le Petit Prince*, and lexical transfer results were found quite encouraging with a recall of 86% and a precision of 71%.

1. Introduction

Lexical transfer from French to English has been widely explored using available electronic resources. However, most of these resources have been exploited with data structures, either close to those of human readable dictionaries, or designed according to a knowledge representation theory allowing the reusability of some properties through computation (e.g. conceptual or UNL graphs, lattices, semantic networks). In such frameworks, word sentence disambiguation (WSD) in translation is a complementary task, and part of a complex module computing the proper candidate term. For instance, if the French word *course* is available, in a given *language resource* (LR) with its English equivalencies *race*, *errand*, and *shopping*, an astute combination of disambiguation procedureshave to be triggered before selecting the appropriate term while translating the following sentences:

1.Les **courses** de chevaux ont lieu les mardis.
*horse **races** occur on tuesdays.*
2.Je l'ai envoyé faire une **course**.
*I send him on an **errand**.*
3. Je dois faire mes **courses**.
*I must go **shopping**.*

The situation is summarized by the following alternative options: If the LR is in the shape of a dictionary, WSD has to rely on sophisticated techniques, and if the LR is a knowledge base, WSD is more obvious, but effort for building the resource is heavy, often leading to partial representations and domain-restricted ontologies.

2. Goal of the paper

This paper proposes a solution to this dilemma in the shape of a trade-off between a dictionary and an ontology, and shows how it might be associated to a translation tool based on morpho-syntactical parsing of the source language.

In this sense, a very interesting resource is the one issued by the Gutenberg project, aiming at providing the Roget Thesaurus of English Words and Phrases (Roget, 1852) as a resource. As a dictionary, the Roget has been very widely used by researchers in Natural Language Processing. But the Roget is also an ontology that covers the whole language. Its design is such that every word of the language is **indexed** by a set of **concepts** i.e. other words chosen as essential notions representing knowledge and meaning. When looking at this, one may find that:

(i) Each word meaning could be represented in this frame as a set of associated concepts, hinting at ideas triggered by this word. Thus, representation is a potential and not a list of particular instances

(ii)Concepts are organized into hierarchies, with a striking familiarity with an ontology. But here, it is the ontology of language, and not of a specific domain

(iii) Concepts are themselves indexed by other concepts.

(iv)The work has been done for most used words of the language, that is, about $50,000$.

(v) Concepts are about $1,000$.

3. Related Works

The idea that an ontology of language could be useful has been already present in works such as (Yarowsky, 1992) (Wilks, 1998) and a few hundred references are nowadays available mentioning the Roget. Other languages have their version of the Roget Thesaurus. Larousse lexicologists provided their main draft for the French language in (Larousse, 1992), followed by an electronic version in 2000. At the same time also, vector-based representations of texts, presented long ago (Salton, 1968) were beginning to regain popularity, after having receded for a long while into a second place position. With the emergence of the Web, offering huge amounts of texts as possible corpora, the Saltonian vector-based approaches were rehabilitated (Salton and MacGil, 1983), and other vector-based techniques such as LSI (Deerwester et all. , 1990) and SVM (Joachims, 1998), appeared and provided successful solutions to Information Retrieval issues. Most works have used vector representation for tasks such as indexation, classification or retrieval. Few have suggested a mathematical representation of dictionaries in the like of vectors. One of the most important ones and probably one of the earliest, is the LDOCE vector representation suggested by (Wilks et al., 1990)) for machine translation. The algorithm calculated a context vector for words (through co-occurrence matrices), and LDOCE concepts were represented as aggregate vectors, thus accounting for polysemy. Other works have followed this approach, but mostly focusing on context vectors in a WSD task : (Niwa and Nitta, 1994), (Inkpen and Hirst, 2003) and (Patwardhan and Pedersen, 2006) with their gloss vectors based on Wordnet semantic relations.

4. Using the Roget as a Semantic Vector Space

Simultaneously with Wilks et al., (Chauche, 1990) presented a semantic representation based on a vector representation of lexical entries, associated with a semantic calculus of a sentence vector. The idea was very simple: why not use a 'Roget-like' set of concepts as the generative family of a vector space, and the Roget indexing system as a vector representation of each word? The author presented his argument using a very early draft of the French Larousse Thesaurus, which at that time was only in paper. In his paper, he showed how to transfer the lexical ontology in a vector space. For instance, the word "today", is indexed by 196.1, 202.3, meaning that the adverb "today" is projected on concepts 196 (PRESENT) et 202 (RECENT). Values after the (".") are morphological indications. Formally, we represent this by the formula:
$\Pi_i(\vec{t}) = 1$ if C_i indexes t, else $\Pi_i(\vec{t}) = 0$, $i \in (1, ...n)$ where n is the dimension of the generative family.
For English, the present version of the Roget gives $n = 1000$. For French, the Larousse thesaurus gives $n = 873$.

4.1. Semantic Calculus of Sentence Vectors

Associated with such a representation, a *sentence vector* is calculated in the space defined hereby as a linear combination of phrase vectors which, in turn, are linear combination of words vectors. The weights associated with word vectors depend on their syntactic role in the sentence, which is parsed and transformed into a constituents tree by the SYGFRAN parser for French (Chauche, 1984). Governing constituents weigh more than governed items. Weights are defined as powers of 2 beginning with 2^0 for the leaf of the most dependent constituent to 2^p, p representing the rank of the highest governing component in the parsing tree(i.e. verb and subject are at highest).
The mathematical representation is the following. Let γ be a parsed phrase. It could be defined as an ordered set of words $w_1, w_2, \ldots, w_n$, represented by their respective vectors $\vec{v_1}$, $\vec{v_2}$,...,$\vec{v_n}$. Let λ be a power of 2 representing the weigh of each word in the phrase tree. The phrase vector, $\vec{\gamma}$, is obtained as the recursive normed sum of:
(1) Words directly belonging to the phrase;
(2) Sub-phrases composing the phrase.
Thus, for every phrase of an i level in the parsing tree (root has got level 0 which is 'highest' and terminal leafs the lowest level, n), we have the recursive formula for $\vec{\gamma} i$:

$$\vec{\gamma_i} = \frac{\sum_j \overrightarrow{(\lambda_j v_{j,i+1})}}{\| \sum_j \overrightarrow{(\lambda_j v_{j,i+1})} \|} \tag{1}$$

Let σ be a parsed sentence. Since σ is a phrase of level 0, let ϕ_j be the phrases of level 1 (directly under the root) composing σ. The formula computing σ is:

$$\vec{\sigma} = \frac{\sum_j \overrightarrow{(\lambda_j \phi_{j,1})_{nor}}}{\| \sum \overrightarrow{(\lambda_j \phi_{j,1})_{nor}} \|} \tag{2}$$

This semantic representation has been used in a classification task of a corpus of $4,844$ press articles in French (Prince and Chauché, 2007), with a precision of 0.85 and a recall of 0.94 in a three-categories classification run.

4.2. Word Contextual Vector for WSD

The preceding formula represents the semantic vector of a sentence and thus, gives the trend that the sentence might impose to the words composing it. Words have multiple meaning therefore, the *contextual meaning* of a word in a sentence reflects the impact the its fellow words in the sentence. This contextual word meaning is represented by a product of the word 'dictionary vector' with its sentence vector. This results in a contextualized word vector which can be formulated as following. Let w_p be a word in the sentence σ_k. Its contextualized vector is:

$$\overrightarrow{w_p/\sigma_k} = \vec{v_p} x \vec{\sigma_k} \tag{3}$$

Where $\vec{v_p}$ is the Roget vector of word w_p. To illustrate this, we will examine vectors for the word *courses* meaning *race* or *errand* or*shopping*, in its potential form, then after processing the three sentences provided in introduction. As a convention, we represent vectors at this stage as a set of the indexing concepts numbers.

5. Building a Roget-Larousse Vector Representation For Lexical Transfer

5.1. Concepts Translation: Bilingual Transformation Family

The original setting of the here-above method was, as mentioned, built for the French version of the Roget, the Larousse Thesaurus which provided an ontology of 873 concepts leading to a vector space of dimension 873. The Gutenberg Project version of the Roget proposed 1000 (the original Roget reached 1043 indexing concepts). So, to built a bilingual Roget-based LR, a simple correspondence of indices was not possible. A mere transformation matrix was not possible either, because the Roget concepts and the Larousse concepts were not translations of each other.
We manually built a set correspondence between the Roget and the Larousse concepts: It has to be done once for all. We present hereby an extract of the first 15 concepts (over 1000). On the left side, the English concepts, and on the right, the French concepts translating them. We used the Hachette-Collins bilingual dictionary.

1: existence:existence,présence
2: inexistence:inexistence,absence
3: substantiality:matérialit,substance
4: unsubstantiality:immatérialité
5: intrinsicality:qualité
6: extrinsicality:accident
7: state:état
8: circumstance:circonstance
9: relation:relation
10: irrelation:indépendance
11: consanguinity:hérédité,filiation,famille
12: correlation:réciprocité
13: identity:identité
14: contrariety:altérité, opposition
15: difference:altérité, différence, diversité

Then, we represented the vector of every French concept in the English space as follows: Let $\vec{FC_1}$ be the vector in the Larousse space of the French concept number j. Let $\overrightarrow{FC_1/Rog}$ be its vector calculated in the Roget (English) space.

$$\overrightarrow{FC_j/Rog} = \Sigma_i(\vec{EC_i}) \qquad (4)$$

where $\vec{EC_i}$ is the vector of the English Concept number i translating the English concept j. In this example, the vector of the French concept 'altérité' in the English space is the sum of the vectors 14, 'contrariety' and vector 15 'difference'. Thus $\overrightarrow{FC_j/Rog}$ is in a space which dimension is 1000. Let us note that the vector is normed afterwards, for commodity sake. We have generated a Bilingual Transformation Family, called BTF, which is now a new generation family in the French space. A first step was to suggest it as a matrix, however, its size was a real liability. So we preferred to use the BTF vectors only at need, when calculating word vectors.

5.2. Monolingual Vector Dictionaries

We had generated beforehand $50,000$ vectors for French words in the Larousse space, corresponding to the French Vector Dictionary (FVD) for our usual needs. Since nothing existed as such in English, we wrote up a routine transforming the textual output of the Gutenberg Project sense indexation into vectors. An extract of this output is provided hereafter.
airship: - ship 273 N. **air wind**: - height 206 Adv. **airy hopes**: - hope 858 N. - hopelessness 859 N. **airy**: - air 338 Adj. - unimportance 643 Adj. - cheerfulness 836 Adj. - levity 320 Adj. - unsubstantiality 4 Adj. - gaseity 334 Adj. so we built up the English Vector Dictionary (EVD) by extracting English concepts numbers. For instance, the word *airy* had the following representation: $(338, 643, 836, 320, 4, 334)$ meaning that in the 1000 space dimension of the Roget, the vector for *airy* had non null components on these 6 concepts. Building the EVD dictionary took some effort, and we came up with about $50,000$ entries, among which noun and adjectival phrases (see for instance *airy hopes*) corresponding to expressions. An extract of the EVD code is provided hereafter.

```
'berth' : Categorie = Nom ;
SV=%VECTEUR(ARITH:189,215,625).
'beryl' : Categorie = Nom ;
SV=%VECTEUR(ARITH:435,847).
'besetting sin' : Categorie = Nom ;
SV=%VECTEUR(ARITH:945).
```

The Category is the POS tag (here 'nom' means 'noun') and SV means semantic vector, and the numbers following ARITH are the non null components Indexes. This follows on for all the EVD entries.

5.3. A French to English Vector Space

We had two options: Either to transfer English words vectors into the French space ($dim = 873$), or to transform French words vectors in the English space ($dim = 1000$). We chose the latter option. We first proceeded by relating the French lexical entries to their English equivalencies, by attributing to these entries the vectors of those equivalencies. Thus, we relate the French word to its English translations provided by the Collins-Hachette Dictionary by enumerating the EVD vectors of the English equivalencies . For instance, for ur sample word, *course*, we had the following extract :

```
'course' : Categorie = Nom
;LemmeAng='errand';
SV=%VECTEUR(ARITH:104,78,264,
267,964,613,143).
'course' : Categorie =
Nom;LemmeAng='journey';
SV =%VECTEUR(ARITH:264,266,302).
'course' : Categorie = Nom;
LemmeAng='rush';
SV =%VECTEUR(ARITH:684,643,367,348,
274,173,171,72).
'course' : Categorie = Nom;
;LemmeAng='race';
SV =%VECTEUR(ARITH:622,708,692,625,75,392,69,
166,274,720,348,350,11).
'course' : Categorie =
Nom;LemmeAng='racing';
SV =%VECTEUR(ARITH:274,347,684,720,840).
'course' : Categorie = Nom;
;LemmeAng='travel';
SV =%VECTEUR(ARITH:266).
'course' : Categorie =
Nom;LemmeAng='stroke';
SV =%VECTEUR(ARITH:731,830,680,655,
626,619,276,550).
'course' : Categorie = Nom;
;LemmeAng=' flight path';
SV =%VECTEUR(ARITH:627,671,302,295,
294,278,260,692). 'course' :
Categorie = Nom; LemmeAng='passage';
SV =%VECTEUR(ARITH:270,680,
593,413,302,260,
191,189,151,144,51,627,267).
'course' : Categorie = Nom;
;LemmeAng='privateering';
SV =%VECTEUR(ARITH:791).
'course' : Categorie = Nom; Nombre =
Plu;LemmeAng='shopping';
SV =%VECTEUR(ARITH:794,795).
```

The **LemmeAng** category gives the English word which vector is here associated to the French string '*course*'. Let us notice that the last description includes *number* (in French Nombre) as a condition. *Course* can only be translated by *shopping* if it is plural in the original language (as *courses*).

6. Lexical Transfer in the Translation System

6.1. Word Equivalency Disambiguation

Finding the proper translation of a word could be seen as finding the closest target-language word to the meaning of the source-language word in the sentence where it appears. In our application, this comes to finding the closest English word (or phrase) to a French word (or phrase) in its sentence. Translated into the vector framework, it means finding the closest English word vector in the French space to the contextualized vector of the word. This induces using a distance between vectors, and one of the most obvious ones is the cosine. The idea is that the English word w_p is the closest translation of the French word m_n (of vector $\vec{m_n}$) in the context of sentence σ_k if the scalar product $\vec{Fv_p}.\overrightarrow{m_n/\sigma_k}$ is the smallest among all others, involving French words vectors in the French space. Note: the scalar product is enough since all vectors are normed when summed.

The procedure is the following.

(1)We transform the contextualized vector of the French word (original dimension=873) into a vector in an English space by multiplying its non null components with the BTF vectors of the indexing concepts. For instance, if a word vector has an activated (non null) concept 8 in Larousse(which is 'altérité') then its vector in English space will activate concepts 14 (contrariety) and 15 (difference), and the original value of its 8th will be pasted to the 14th and 15th components in the 1000 dimension space. The calculus of the French to English vector of a French word v_p which contextualized vector is $\overrightarrow{v_p/\sigma_k}$ after semantic calculus, is the vector $\overrightarrow{FE(v_p)}$, which is obtained as follows:

$$\overrightarrow{FE(v_p)} = \overrightarrow{v_p/\sigma_k} * BTF \tag{5}$$

which transforms the indexing concepts of the French word into their English related concepts.

(2) We will compare this vector its FEVD fellows, i.e. to the vectors of the entries which have the same character string. In our example, we will compare the contextualized vector of *course* in sentence 1 (*Les courses de chevaux ont lieu tous les mardis*), with the 11 entries in the FEVD base by calculating the cosine.

(3) We retrieve the entry which has the smallest cosine value. In sentence 1, we have two close values (very slightly different) of the cosine with the entries which have 'race' and 'racing' as translations, with a very slight preference for 'race'.

6.2. Integrating into the Translation System

Our translation system, the SYGFtoE prototype, is an asymmetrical translator which relies on a heavy parsing of the source language, French, and a light generation of the target language, English. The procedures runs as such:

1. The text in French (as long as the writer wishes) is sent first to the SYGFRAN French parser. It generates a constituent tree, and calculates dependencies such as object and sentence complements and noun phrase modifiers.

2. A Contextualized vector for each word is calculated according to its position in the French tree, using the FVD as a primary LR (as described in subsection 4.2).

3. A reading of the FEVD set gives the English words as equivalencies

4.The cosine of FEVD selected vectors with the contextualized vector is calculated. The closest vector is chosen.

5. The The SYGFRAN tree structures coming as an output, are decorated with English words corresponding to the chosen vectors.

The lexical transfer being ended, the system undergoes the syntactic transformation of the sentence.

7. Results

To test the validity of the approach, we have set a small experiment with a $20,000$ words extract of literary text , the *Little Prince* of Antoine de Saint-Exupery, for which we had the following properties:

1.The text contained several polysemous words and thus was subject to a WSD in a classical framework.

2. It was totally correctly parsed by SYGFRAN. SYGFRAN's precision/recall is 0.34 on current input text. This means that it 34% of the parsed sentences are correctly parsed, knowing that all sentences are parsed. For the *Little Prince*, SYGFRAN has a 100% precision/recall value.

3. We had an English version of the text provided by a human translator.

7.1. Untranslated words

This gave us the best case situation to observe the lexical transfer accuracy. We obtained a recall of 0.86 and a precision of 0.71. 10% of the words were not translated at all. The errors related to that were the following.

- Spelling errors in the text and transcription errors in FVD (dictionary entries misspelled).

    1. Examples of typos and abbreviations in the text: In our extract, which is the result of a scan, we had abbreviations or errors such as: *no un* instead of *numéro un*, *a* instead of *ça* , *fleure* instead of *fleur*, which by the way introduced an error in parsing, because *fleure* exists as a conjugated form of the verb *fleurer* which is seldom used.

    2. Examples of FVD entries misspelled:
       asteroide instead of *astéroïde*
       aditions instead of *additions*.
       Since our FVD is semi-automatically generated, some of our entries had typo errors, or diacritics were missing.

- The remaining untranslated words were missing entries in FVD. Naturally proper names are not translated but we had entries such as:
    1. *Vénus, Terre*, *Afrique* for which the first letter is a capital, indicating a proper name.They had to be translated into *Venus*, *Earth* and *Africa*. In the same field, there were several entries with "un Monsieur ' (a gentleman), and the capital letter within the sentence drove the system to assimilate the word *Monsieur* to a proper name. We had the same problem with the main characters, the Prince (but here since it is the same in English and French it did not matter), the flower (la *Fleur*), the fox (le *Renard*), which were all written with a capital first letter. We had to transform afterwards several words by changing the capital letter into a normal one.
    2. But also really missing words such as *brindilles* (meaning *twigs*), *ombrageuse* (meaning either *shadowy* or *irritated*) which we indexed afterwards, thanks to the defaults revealed by parsing results.

7.2. Translation Differences and Errors

For translated words, 21.5% were different in the human translation. A change of style using idiomatic expressions, rewriting the sentence completely, or choosing a particular synonym closer to the literary style, were the main reasons for this difference. A third of these 21.5% (so around 7% of the whole text, adding up to approximatively 140 words or expressions) were akward or inconsistent translations. Cases of inconsistency were few: 49 words over 20,000 as a whole, and 35% of the 140 'wrong' translations were truly inappropriate. For the remaining ninety words, we had awkward translations for words like *modeste*, translated into *modest*, where the best sense was *humble*, *vanité* translated by *uselessness* in a context where both *conceit* and *futility* were possible translations, the latter being a possible synonym of *uselessness*.

8. Conclusion

Results described in the preceding section are encouraging. Although our corpus is not significant in volume, it was a typical concentration of several interesting issues in WSD and thus in lexical transfer. To export this experiment into a broader frame, one will need to:

- Be sure that all sentence in the source language (here French) are correctly parsed. If not, then WSD cannot be granted since contextualized semantic vectors cannot be correctly computed. Translation will proceed on a wrong base, and it might probably lead to a bad final choice. Since parsing precision is only 34% when confronted with a randomly chosen corpus, this is an important liability.

- A mitigated solution to this problem (apart from increasing the parser precision) could be to provide the first entry in the FEVD base, and to highlight it to drive the human user's attention on the fact that he/she must possibly correct the translation. In other words, treat the words as if it had only one possible equivalency (the first one). This has worked for our translation prototype in other contexts (Bonnin and Prince, 2007), in which, by chance, the first equivalency was the good one in quite a few cases. By attracting the user's attention to the item, we believe that it will help him/her focus on translating manually the items if he/she thinks the result inappropriate.

- Increase all our bases. With 50,000 entries, we do not cover all of either French or English languages. If one has to move to specialized domains, then a need for terminological bases and/or bilingual terminology will soon appear.

Nevertheless, the idea that WSD could be more parsing related then knowledge structure related, which is central in this paper, has to be retained as feasible and useful. Generating FVD and EVD was easily done, thanks to the Gutenberg Project and to Larousse. Producing the FEVD data did not require more than a few months effort. With these three dictionaries, we had a basis which could grow at any moment, including new vectors and entries. Calculating contextualized vectors is simple. Calculating the proper equivalency does not require deep intelligence, only a few cosines then choosing the best candidate. The liabilities are those which have been described above. But they can be overcome either by finding other criteria to calculate semantic vectors, or by improving parsing, which is on the way.

9. References

Bonnin G. and V. Prince. 2007. Emphasizing Syntax for French to German Machine Translation. Proceedings of the Seventh Symposium on Natural Language Processing, Pattaya, Thailand. pages 12-20.

Chauché, J. 1984. Un Outil Multidimensionnel d'Analyse du Discours. *Proceedings of COLING84*, Stanford, California.

Chauché J. 1990. Détermination sémantique en analyse structurelle : une expérience basée sur une définition de distance *.TA Information* vol 1/1, p 17-24.

Deerwester S. and S. Dumais, T. Landauer, G. Furnas, R. Harshman, *Indexing by latent semantic analysis.* In Journal of the American, Society of Information science, 1990, 416(6), p 391-407.

D. Inkpen and G. Hirst. 2003. Automatic sense disambiguation of the near-synonyms in a dictionary entry. *Proceedings of the 4th Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2003)* , pages 258267

Joachims T. 1998. Text categorisation with support vector machines : learning with many relevant features *Proceedings of ECML-98, 10th European Conference on Machine Learning* pages 137142 .

Larousse. 1992. *Thésaurus Larousse - des idées aux mots, des mots aux idées.* Ed.Larousse, Paris.

Y. Niwa and Y. Nitta. 1994. Co-occurrence vectors from corpora versus distance vectors from dictionaries. *Pro-*

ceedings of the Fifteenth International Conference on Computational Linguistics. pages 304309.

Patwardhan Siddharth and Ted Pedersen. 2006. Using WordNet-based Context Vectors to Estimate the Semantic Relatedness of Concepts . *Proceedings of EACL 2006*.

Prince V. and Chauché J.. 2007. Classifying texts through natural language parsing and semantic filtering . *Proceedings of Third International Language and Technology Conference*.

Roget P. 1852 *Thesaurus of English Words and Phrases* Longman, London.

Salton G. 1968. *Automatic Information Organisation and Retrieval.* McGraw-Hill New York.

Salton G. and M. J. MacGill. 1983. *Introduction to Modern Information Retrieval.* McGraw-Hill, New-York.

Y. Wilks, D. Fass, C. Guo, J. McDonald, T. Plate, and B. Slator. 1990. Providing machine tractable dictio- nary tools. Machine Translation,. Volume5, pages 99154.

Wilks, Y. 1998 Language processing and the thesaurus. *Proceedings of the National Language Research Institute*. Tokyo, Japan.

Yarowsky D. 1992. Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora *Proceedings of COLING92*.