Tools for Sublanguage-Based Semantic Knowledge Acquisition from Corpora

M. Victoria Arranz* Ian Radford† Sofia Ananiadou‡
Jun-ichi Tsujii§

Centre for Computational Linguistics, UMIST,
PO Box 88, Manchester M60 1QD, England

Abstract

This paper describes the implentation of a KA tool kit for use with sublanguage-specific corpora. The key idea of KA as an evolutionary process is discussed in detail. Special attention is paid to the system's attempt to avoid the pitfalls faced by purely statistical KA processes. The system relies on a set of interactively linked subprocesses and on a central knowledge base which stores all acquired knowledge.
Key Words : Knowledge Acquisition (KA), Sublanguage, Corpus-based Linguistics, Domain Ontology, Statistical Methods, Dynamic Context-Matching Techniques.

1 Introduction

Difficulties in current NLP applications are mainly due to the fact that the knowledge to be encoded in such application systems happens to be unknown. Fortunately, as can be seen in [Grishman and Kittredge, 1986], many of the language processing problems encountered are effectively restricted to the particularities of the language in a specific domain, where this knowledge can be more easily obtained. The variety of language used in a certain domain is much smaller than the whole language as well as more clearly systematic in structure and meaning. Bearing these considerations in mind, we selected for our research a highly specific corpus, such as the Unix manual.

Preparation of this domain-specific knowledge for a NLP application, however, is time-consuming and not easy, due to the complexities of the task caused by the non-trivial relationship between the ontological knowledge and the actual language usage. The necessity for systematic methodologies of knowledge acquisition, which are properly supported by a set of utility programs, has been emphasised by several authors [Tsujii *et al.*, 1992] and

*victoria@ccl.umist.ac.uk (Researcher sponsored by the *Departamento de Educación, Universidades e Investigación* of the Basque Government, Spain)
†ianr@ccl.umist.ac.uk
‡effie@ccl.umist.ac.uk
§tsujii@ccl.umist.ac.uk

[Mitamura *et al.*, 1993]. In particular, this research is interested in discovering a systematic method for the acquisition of sublanguage-specific semantic knowledge and capable of adapting the NLP system to different subject domains.

While researchers on purely statistical methods, such as [Church *et al.*, 1991] and [Brown *et al.*, 1991], have shown promising results for large corpora, they fail to produce useful results for practical NLP systems. The tool kit [ϵ] being developed at CCL aims at upholding the principles of KA as an evolutionary process, i.e. by building the knowledge evolutionarily with the minimum human intervention and with statistical analysis techniques, as well as carrying out KA from relatively small corpora. The size of the corpus we have been working with is of about 100,000 words.

In this paper, we describe work on an iterative and modular approach to statistical language analysis, where the knowledge acquired is stored in a central knowledge base, which is shared and easy to enhance and access by all subprocesses in the system.

2 Transparency in Semantic Clustering

Following the research initiated in [Arranz, 1992] and based on the Epsilon system described in [Tsujii and Ananiadou, 1993], this tool kit is being designed with the aim of providing solutions to the defects of statistical semantic clustering techniques, i.e. opacity of the process and insufficient data.

Although quite a few statistical programs have been developed so far for the discovery of semantic clusters from corpora, none of them seems to be sufficiently effective for practical applications. Statistical methods usually require very large corpora to obtain reasonable results, which is highly unpractical and often unfeasible. These techniques can only generate such results for a small portion of words with high frequency of occurrence [Church and Hanks, 1989].

If the language in a particular subject domain is taken into account, the chances of having access to a very large corpus are very limited. Due to the fact that regularities at the ontological level in a given sublanguage are not directly manifested in the surface textual forms, their discovery generally requires a larger corpus than the one we would need to work with at a parts-of-speech level. When statistical methods are involved in the discovery of ontological knowledge, a discovery of morpho-syntactic regularities in a text and reduction of the complexity of the text is implicitly or explicitly carried out.

We have to take into consideration, though, that the sizes of corpora normally available for development of actual application systems do not match the complexity of the requirements of the task. Therefore, one usually resorts to either working on a structurally annotated corpus or applying human intervention to the interpretation of the results. The cost of annotation makes these methods unfeasible or less attractive for practical purposes.

To the above mentioned difficulty of insufficient data, one should add the fact that statistical processes are completely opaque to the human specialist.

Due to their black box nature, judging whether intuitionally uninterpretable results reflect actual language usage in the domain, or are simply errors due to the insufficient data, causes great difficulty. Consequently, they either revise the results to meet their intuition or accept the results without revision. In either case, one can hardly claim that human intuition and corpus analysis by programs play a complementary role in KA.

On the other hand, Epsilon's idea of KA as an evolutionary process avoids the problems above by achieving the following:

- **Stepwise acquisition of semantic clusters.** In contrast with the single shot techniques presented by the purely statistical approach, our system acquires knowledge as a result of stepwise refinement, i.e. by allowing the specialist to inspect each cycle of new classes being created. The different utility programs of [ϵ] propose hypotheses of new pieces of knowledge, which are checked by the specialist.

- **Design of robust discovery methods.** Since statistical programs are vulnerable, especially at the early stages of knowledge acquisition when the complexity of a text is still too high, we find crucial to reduce the complexity of the text by using more robust techniques. These techniques should be able to cope with words with low frequency of occurrence, which are highly problematic for the set of statistical programs to be applied.

- **Inherent links between acquired knowledge and language usage.** The non-trivial nature of the mapping between the domain ontology and the language usage is another major cause for opacity. Even when working on a restricted domain, we often encounter cases in which a word denotes several different ontological entities of the domain, or conversely, one entity is denoted by different words. If any of these happen, [ϵ] maintains a series of pseudo-texts (cf. section 3) and their relationships with acquired knowledge, so that the specialist can freely see the mutual relations and hence understand why certain semantic classes are formed and at which stages, etc.

- **Effective minimum human intervention.** The specialist's task is simplified by the fact that KA proceeds gradually from concrete pieces of knowledge to abstract ones. This makes it much easier for specialists to interpret the proposed hypotheses and make judgements based on their intuition since, at each stage, they can examine in detail the reasons why the programs generate such hypotheses in terms of both their textual occurrences and relationships with previously acquired knowledge. It should be pointed out that even if human intervention is inevitable (as has also been emphasised by Iris Arad [Arad, 1991] in her quasi-statistical system), and can take place freely at any stage of the process, it is systematised and it applies locally, wherever required by the process.

- **Cost-effectiveness.** A text is annotated or described in a stepwise manner to the extent that it is useful for the purpose of the current

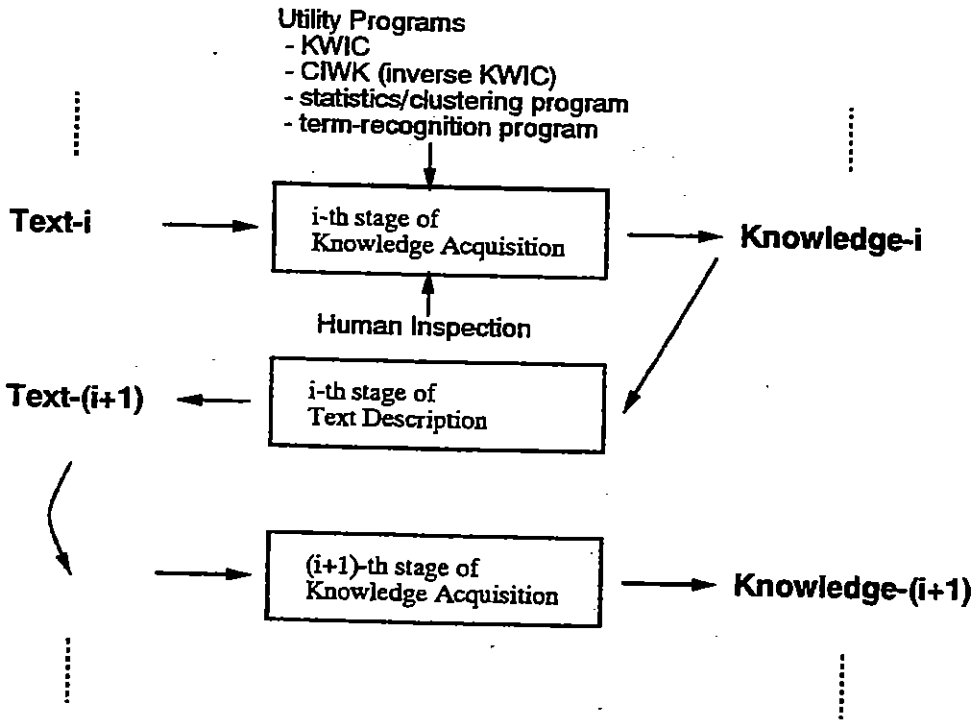

Figure 1: General Scheme of KA as an Evolutionary Process

acquisition of ontological knowledge. There is no need for a full annotation if not specifically required by the process.

3 Replacement Process and Hierarchy of Pseudo-Texts

The general idea of KA as an evolutionary process is illustrated in Fig. 1 [Tsujii *et al.*, 1991]. Application of utility programs to Text-i and human inspection of the results yield the next version of knowledge (the i-th version), which in turn is the input to the next cycle of KA. As shown in Fig. 2 [Tsujii *et al.*, 1991], the i-th version of text description is also a text-like object which shows a lesser degree of complexity than the previous pseudo-text.

The resulting pseudo-texts present the same type of data structure as other texts: an ordered sequence of words, including here pseudo-words as well as ordinary words. Such pseudo-words denote, for example:

1. semantic categories to which the actual words belong,
2. words with part-of-speech information,
3. single concept-names corresponding to multi-word terms,
4. disambiguated lexical items, like in [Zernik, 1991].

As regards the compatibility of the newly created pseudo-texts with the existing utility programs, these can be applied to a pseudo-text as well as to

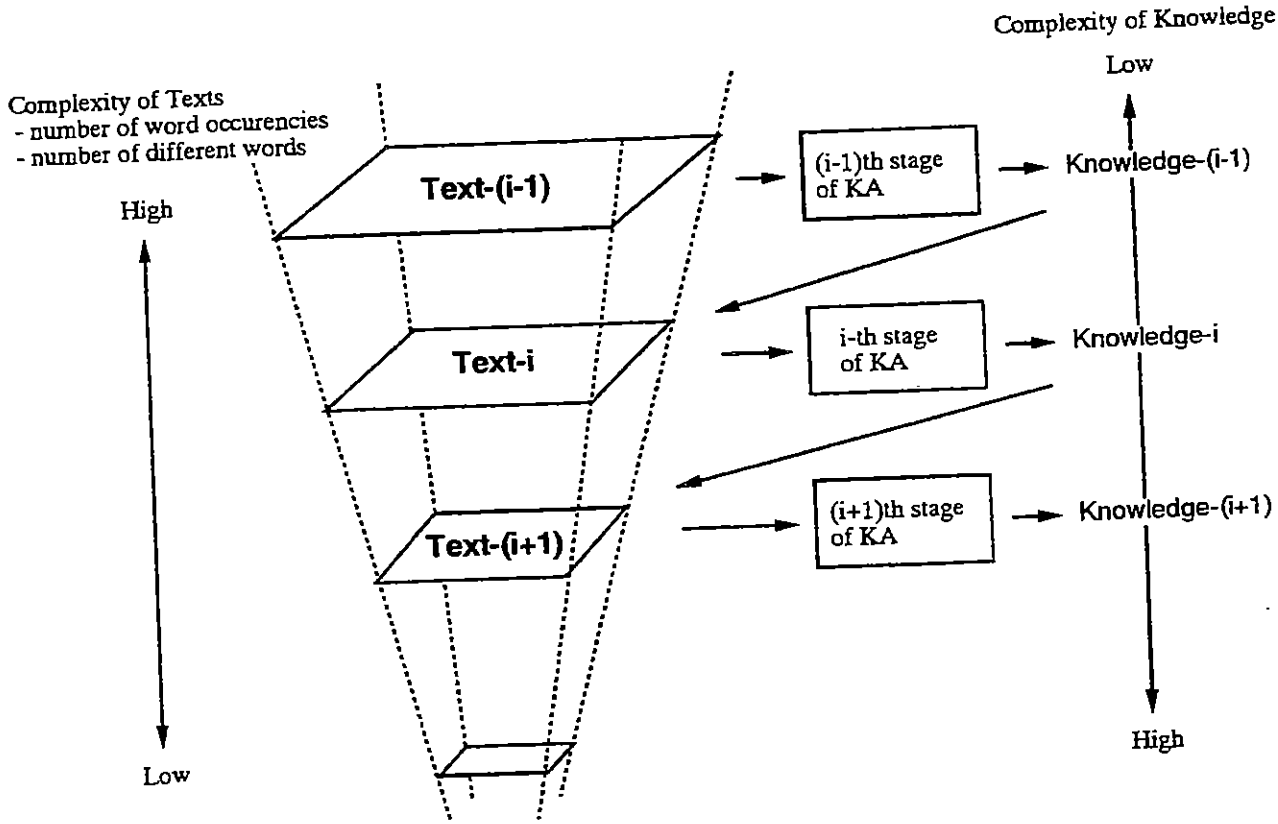

Figure 2: Hierarchy of Pseudo-Texts

an ordinary text, without any need for alteration either of the input data or of the tool itself.

The current version of [ϵ] consists of:

1. Central knowledge base which stores all the relationships among words and pseudo-words obtained during the knowledge acquisition process.
2. Record of the hierarchy of pseudo-texts created as well as the relationships among these pseudo-texts, in terms of replacements or clusterings taking place within the different pseudo-texts.
3. Two main subprocesses, *Compound* and *Classify*, consisting of a series of utilities (cf. following sections), and which generate hypotheses of multi-word expressions and semantically-related term clusters, respectively.

4 Knowledge Acquisition Process

4.1 Preprocessing of the corpus

The initial version of the system did not include the tagging stage. Experimentation on to what extent the extraction of meaning from text with our tools and without any type of annotation in the corpora was intended. Nevertheless, once the *Classify* subprocess was put into practice, it was observed

that the discovery of knowledge at the ontological level was closely related, though indirectly, to the surface textual forms, therefore requiring consequent study of syntactic information in the corpus. The results obtained after several cycles of [ϵ] were promising, but still not acceptable, in terms of the great confusion at the replacement stage. Due to the fact that no part-of-speech information was provided, the replacement program was carrying out illegitimate substitutions which were causing serious incoherence in the resulting pseudo-texts.

After providing the sample text with POS information, the set of candidates for semantically related clusters was much more accurate and the wrong replacements of mixed syntactic categories stopped taking place. Further credit needs to be given too to this corpus annotation, since it allowed us to establish a *tag compatibility set*. This contributed to recover those hypotheses posed for replacement but which presented different syntactic categories and had been previously crossed out from substitution. Each line of the set represents a different group of part-of speech markers that can be interchanged without affecting the coherence of the text. One example of a line in such set looks as follows : **JJ JJR JJS VBN**

The tagger chosen for our work is E. Brill's [Brill, 1993], which is language-independent[1] and permits training. This property of being trainable is of great help when new unannotated text belonging to the same sublanguage is to be used. It will just be added that the accuracy of the tagger for our current corpus oscillates between 88.89% before any training takes place[2], and 94.05%, with a single pass of training.

4.2 System Framework

A number of separate subprocesses are involved during the processing of each pass of the system. These subprocesses (detailed below) rely upon the iterative application of simple analysis tools, updating a central database with the knowledge acquired at each stage. The resulting modular system is of a simple-to-maintain and enhance nature. At present, there are three major processes involved in the knowledge acquisition task:

- *Compound*: This tool fulfills the search for those compound or multi-word structures within the text (or pseudo-text) that can be ranked as single ontological entities.
- *Classify*: This process deals with the grouping of the lexical annotated entities in the sample text into semantic clusters.
- *Replacement*: This module treats the reduction of the complexity of the text. By means of its utility programs, it carries out the replacement of all selected clusters and compounds within the corpus.

[1] An application of the system to both Spanish and Japanese is already planned, therefore requiring tools multilingually applicable.

[2] Which is quite impressive considering the specificity and technicality of the text.

4.3 The *Compounding* Module

As with the rest of the subprocesses, one of our priorities with this module is to ensure that it follows the system's interactive framework. In the first stage of *Compound*, the corpus is analysed using a simple grammar and the structures of potential compound terms are described. This grammar is based upon pairs of words where the second word is a noun and the first is one of the class **Noun, Gerund, Adjective**. Therefore, a single pass can only determine two-word compounds, requiring multiple passes if longer compounds are to be discovered. Once the set of potential compounds is established, they are filtered by simply ensuring that they occurred in the text more than once.

The system then prioritises the remaining candidates by calculating the *mutual information* [Church and Hanks, 1989] of the pair. It was expected that functional words such as *the/DT*, which happen to occur with very high frequencies in the sample text, would not be considered as elements of compound expressions and would be discarded immediately. On the other hand, the adjective *standard* would achieve a high *mutual information* score since it occurs very frequently, but with a very restricted set of nouns (such as *input* or *error*).

Once the filtering by means of *mutual information* has been performed and the resulting set of compound term candidates has been verified by the human expert, each occurrence of every accepted compound is replaced by a single token (or pseudo-word in future runs of the programs) within the corpus. The token formed by the compounding of the adjective *standard/JJ* and the noun *input/NN*, for instance, would be the following:

```
compound(standard/JJ^/^input/NN)/NN
```

The performance of *mutual information* in the *Compound* process has proved to require a great deal of human interaction since, only a 40% of the potential compounds discovered turned out to be positive cases. Most of the negative cases are *Adjective Noun* and *Gerund Noun* pairs. The difficulty, thus, entailed the distinction between general language *Adjective Noun* and *Gerund Noun* syntactic constructions and domain-specific pairs. This problem is mainly due to the fact that some of the compounds are only occurring two or three times in the whole corpus, which means that the frequency measures used to calculate the *mutual information* scores are noisy and lead to a fairly arbitrary result.

A way to measure the specificity of our compounding candidates is by means of a large corpus of general language. The *LOB* corpus has been used here[3] to compare the frequency count of adjectives and gerunds from the *sublanguage* text being analysed with the frequency counts of the same words in the general language corpus. Using the formula shown in equation 1, the *specificity coefficient* for the mentioned terms is calculated, which indicates how specific they are to the sublanguage.

[3] "The Lancaster-Oslo/Bergen (*LOB*) Corpus is a million-word collection of present-day British English texts"[Johansson and Hofland, 1989].

$$Specificity(w) = \frac{f_{corpus}(w) - f_{LOB}(w)}{f_{corpus}(w)} \quad (1)$$

For any given *Adjective Noun* or *Gerund Noun* pair, the range of *specificity* values is based upon the value zero indicating that the word occurs with identical frequency in both the sublanguage and general language. A value of 1.0 implies that the word is unique to the sublanguage while negative values represent a word which is more common in general language than in our domain specific corpus. By using a threshold of 0.9 on adjectives, the accuracy of the compounds generated has improved from 40% to 64%, filtering out almost a 50% of the errors to be detected by the specialist.

Once the compound terms are verified, they are replaced by single tokens of a pseudo-word, *compound identifiers* such as *Compound56/NN*. This facilitates the storage of the information in the central knowledge base as well as making it more accessible for the subprocesses. In later cycles of the KA process, *Classify* and *Compound* will treat this *Compound56/NN* as an ordinary word with a *NN* part of speech. Meanwhile, the knowledge base of [ϵ] keeps a record of the information relating to this token.

4.4 The *CIWK* Context Matching Module

This module represents the first stage in the *Classify* subprocess, where the concordance program *CIWK* (or *Inverse KWIC*) [Arad, 1991] is used to identify words sharing a common local context. *CIWK*'s output looks as follows, where $ stands for the place in the context where the selected terms occur:

```
BEGIN/NNP ;END/NNP ; # special/JJ pattern/NN $ may/MD be/VB
field/NN ;record/NN ; # the/DT output/NN $ separator/NN (/(
```

Once the output list of semantic clusters is checked, the corpus is updated, with all the words within each cluster being replaced by the first instance of that group. Consequently, in our example, all occurrences of *END/NNP* and *record/NN* would be replaced by *BEGIN/NNP* and *field/NN*, respectively. A relatively small context (two words preceding and two succeeding the word being considered) has been selected after examining several possibilities, so as to produce a larger set of hypotheses. With this context, a list of around 700 classes has been obtained. Among these, though, an important number of ambiguous cases takes place.

5 Performance of Modules and Further Development

5.1 Performance Evaluation

When [ϵ] is applied to the Unix manual corpus we are using, both currently working subprocesses *Classify* and *Compound* obtain very interesting results. Although more work has been done into *Compound*, *Classify*'s module *Inverse KWIC* produces a rather large list of semantic clusters, given a [2 2]

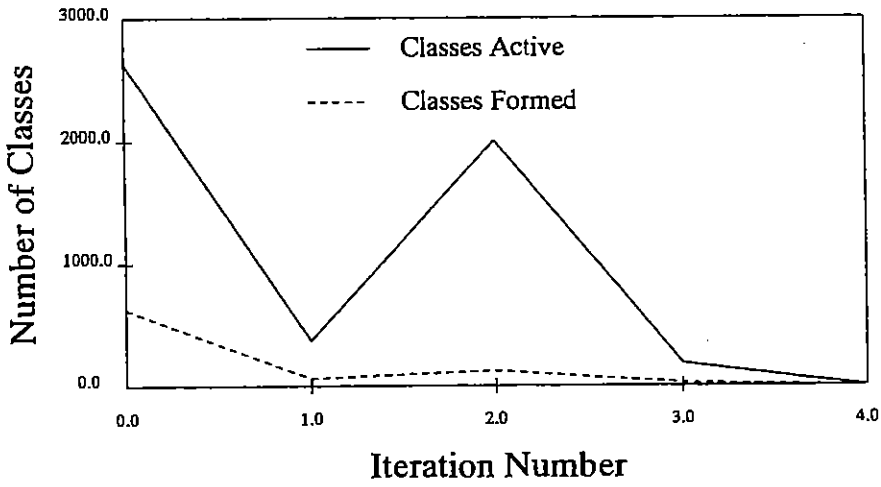

Figure 3: Semantic Clustering Results

contextual size parameter. It provides around 700 classes, out of which many will be later discarded, but of which, an interesting number presents crucial ontological and contextual features. Work is currently taking place in the filtering of all these selected clusters by means of a technique called *Dynamic Alignment* [Somers *et al.*, 1994] and which has already provided us with very promising disambiguation cues (cf. section 5.2 for preliminary results). This should help to reduce ambiguity as well as human intervention. The number of semantic classes formed and the number of actual instances of each class are shown in Fig. 3, and the corresponding results for compounds in Fig. 4.

Regarding the performance of *Compound*, the first stage of the process has produced around five hundred hypotheses of multi-word expressions. Since the vast majority of these are irrelevant, the syntactic filtering carried out by our grammar (cf. section 4.3) leaves us with around 70 candidates. Out of these 70, 45 present *Noun Noun* pairs, and the remaining 25 are *Adjective Noun* or *Gerund Noun* pairs. As already discussed, it is the latter type of compounds which creates the greater difficulty, obtaining only about 40% of actually correct compounds, compared to the 85% in the former type.

The filtering of this disappointing 40% by means of the *LOB* corpus improves performance to a promising 64%. Over all, then, the final percentage of correct hypotheses of compounds adds up to be 77.5%, just after the first pass. The performance of this last stage of filtering is currently rather limited, though, considering the fact that the statistics regarding the word frequencies in the *LOB* corpus do not take POS information into account.

The iterative nature of [ϵ] is a strong point in this KA process. Firstly, it enables common statistical measures (such as *mutual information*) to be performed. Initially, the size of the corpora may result relatively small to find elements sufficiently frequent so as to allow such methods to be statistically valid. Nevertheless, as each pass occurs and the corpus reduces complexity, the analysis of the less frequent elements of the text becomes possible.

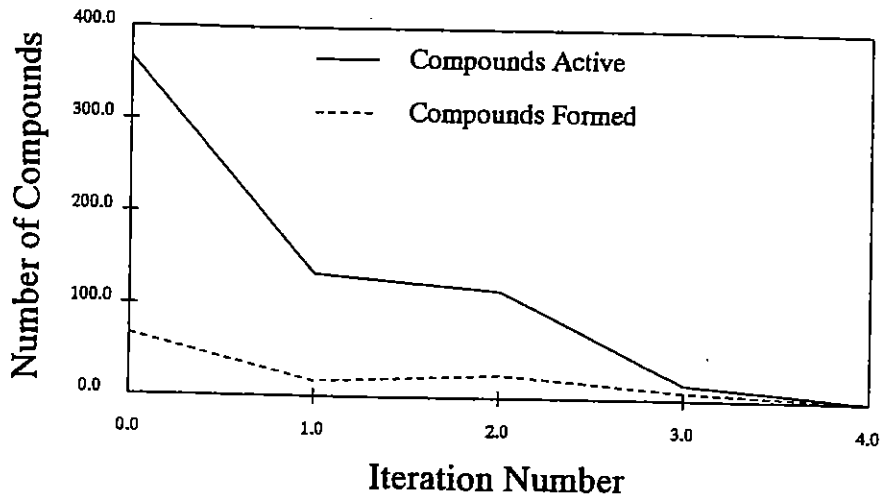

Figure 4: Compounding Results

Secondly, even though many correct multi-word expressions are left undetected in the first cycle of the process, they will be detected in later cycles of acquisition, together with multi-word expressions longer than two words. This will even allow, in later passes, to form clusters of semantically related compounds, which present different word lengths. Thirdly, only the knowledge of which we may be most confident is acquired at each pass. Then, this knowledge is used to reduce the complexity of the text so that the next level of knowledge can be acquired more accurately.

The interaction of subprocesses through cycles is also of interest. While *Classify* generates a higher number of semantic classes in later cycles, which contain pseudo-words of semantic clusters formed in previous cycles, *Compound* and *Classify* interact with each other to obtain desirable results.

We conclude this section by briefly referring to the important role that the central knowledge base plays within the framework of [ϵ]. The modular approach of the system together with the accessibility of the stored acquired knowledge provide an easy way to update and improve the knowledge base, as well as an opportunity to add new modules within the framework.

5.2 Semantic Clustering Disambiguation through Dynamic Context Matching

As mentioned above, work on a filtering module for *Classify* is currently being undertaken. The *CIWK* algorithm is very inflexible in that it can only match up exactly matching contexts. In practice, two words that should be considered as part of a semantic cluster may have very similar contexts, but varied slightly due to one of several different reasons.

Dynamic matching [Somers *et al.*, 1994] is a technique which allows us to compare the degree of similarity between two contexts. This is a much more flexible approach than simply determining that the contexts are, or are not, identical. The technique involves discovering all of the potential matches

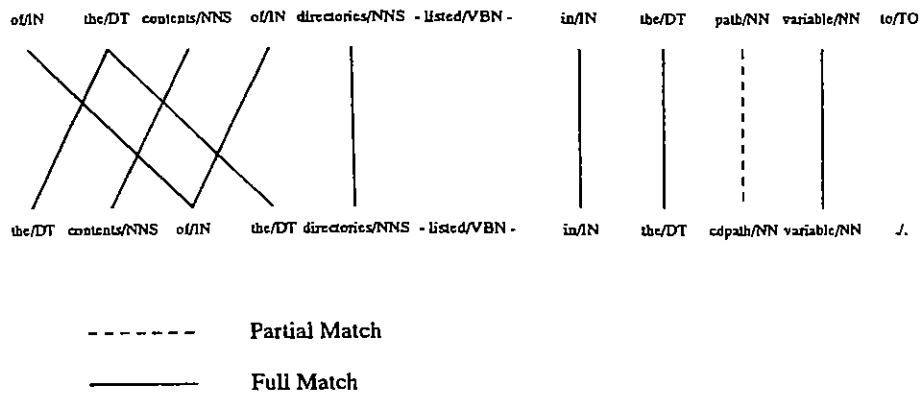

Figure 5: Example match between two contexts

between individual words, each match having a value attached indicating how important it is. The task then becomes one of calculating the set of matches which produces the highest total match strength. This highest score is the value attributed to the pair of contexts to indicate how similar they are.

An example of the possible word matches for one pair of contexts is shown in Fig. 5. From this, the maximal set is chosen, given the constraint that the individual matches are not permitted to cross (thus forbidding a set in which both the first *of/IN* of each context are matched at the same time as the first *the/DT*).

The fundamental idea behind dynamic matching is that a set of words will be provided, from which the dynamic matching program will find the strongest semantic cluster (consisting of context instances from any or all of the supplied words). For each pair of contexts in which any of the supplied words are found in the corpus, the best match value is calculated. This results in a *correlation matrix* being formed such as the one below:

```
% dynamic discussed/VBN listed/VBN +5 -5 < corpus
Post context length set to 5
Pre context length set to 5
CIWK data read. 9 records found.
    0  1  2  3  4  5  6  7  8
-----------------------------
0 :    5 10  7  6  8 10  7 10
1 :       8  7  8  3  5  5  4
2 :         27  7 11  9  5  4
3 :             6  8  6  4  4
4 :               14  9  6  4
5 :                   14  1  3
6 :                       4  5
7 :                          5
8 :
```

From the above matrix, the group which represents the most concrete

semantic cluster must be determined. This is currently performed by a simple clustering algorithm which operates in the following manner:

1. The pair of contexts which has the highest correlation are chosen as the core of the cluster.

2. Consider each remaining context in turn: if that context has a correlation value of above a certain threshold, with more than half of the contexts in the current cluster, then add it to that cluster.

3. Repeat step 2 until no more contexts are added to the cluster.

The current value for the threshold, as well as the strength of the full and partial matches are still under consideration. It should be noted that this method, as well as being more flexible than *Inverse KWIC*, implicitly solves the ambiguity problem detailed above by filtering all the clustering candidates presented during the *Classify* subprocess. By treating each instance individually rather than looking for words which match in general, the different meanings of words should be discovered. In addition to the elements forming the semantic cluster, this matching technique also provides the contexts containing the necessary ontological knowledge to disambiguate between the components of the different clusters.

6 Concluding Remarks

This paper has presented the reseach currently taking place on the implementation of a KA tool kit. This tool kit aims at providing solutions to the problems encountered by purely statistical techniques: opacity and insufficient data. Our modular interactive approach deals with these problems by proposing a stepwise acquisition of semantic clusters, a design of robust discovery methods and inherent links between acquired knowledge and language usage.

As previously discussed, when statistical techniques such as *Mutual Information* are applied to our rather small sample text, further support is required from other tools to reach satisfactory results. Hence, our idea of KA as an evolutionary process, proves to be of considerable efficiency for our intended knowledge extraction process from relatively small corpora.

References

[Arad, 1991] I. Arad. *A Quasi-Statistical Approach to Automatic Generation of Linguistic Knowledge.* PhD thesis, CCL, UMIST, 1991.

[Arranz, 1992] V. Arranz. *Construction of a Knowledge Domain from a Corpus.* Master's thesis, CCL, UMIST, 1992.

[Brill, 1993] E. Brill. *A Corpus-Based Approach to Language Learning.* PhD thesis, University of Pennsylvania, 1993.

[Brown *et al.*, 1991] P.F. Brown, S.A. Della Pietra, V.J. Della Pietra, and R.L. Mercer. Word Sense Disambiguation Using Statistical Methods. In *Proceedings of ACL*. Association of Computational Linguistics, 1991.

[Church and Hanks, 1989] K. W. Church and P. Hanks. Word Association Norms, Mutual Information, and Lexicography. In *Proceedings of ACL*, pages 76–83, Vancouver, 1989. Association of Computational Linguistics.

[Church *et al.*, 1991] K. W. Church, W.A. Gale, P. Hanks, and D.M. Hindle. Using Statistics in Lexical Analysis. *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*, 1991.

[Grishman and Kittredge, 1986] R. Grishman and R. Kittredge. *Analysing Language in Restricted Domains: Sublanguage Description and Processing*. Lawrence Erlbaum Associates Publishers, London, England, 1986.

[Johansson and Hofland, 1989] S. Johansson and K. Hofland. *Frequency Analysis of English Vocabulary and Grammar: Based on the LOB Corpus. Tag Frequencies and Word Frequencies*, volume 1. Clarendon Press, Oxford, England, 1989.

[Mitamura *et al.*, 1993] T. Mitamura, E.H. Nyberg, J.G. Carbonell, and et al. Automated Corpus Analysis and the Acquisition of Large, Multi-Lingual Knowledge Bases for MT. In *Proceedings of TMI-93*, Kyoto, Japan, 1993.

[Somers *et al.*, 1994] H. Somers, I. McLean, and D. Jones. Experiments in Multilingual Example-Based Generation. In A.I.C. Monaghan, editor, *Proceedings on the 3rd Conference on the Cognitive Science of Natural Language Processing*, Dublin City University, Dublin, Ireland, July 1994.

[Tsujii and Ananiadou, 1993] J. Tsujii and S. Ananiadou. Epsilon [ϵ] : Tool Kit for Knowledge Acquisition Based on a Hierarchy of Pseudo-Texts. In *Proceedings of Natural Language Processing Pacific Rim Symposium*, pages 93–101, Fukuoka, Japan, December 1993. Information Processing Society of Japan.

[Tsujii *et al.*, 1991] J. Tsujii, S. Ananiadou, J. Carrol, and S. Sekine. Methodologies for Development of Sublanguage MT System II. Technical Report No.91/11, CCL UMIST, 1991.

[Tsujii *et al.*, 1992] J. Tsujii, S. Ananiadou, I. Arad, and S. Sekine. Linguistic Knowledge Acquisition From Corpora. In *Proceedings of FGNLP*, Manchester, England, 1992. CCL, UMIST.

[Zernik, 1991] U. Zernik. Train1 vs. Train2: Tagging Word Senses in Corpus. In *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*. Lawrence Erlbaum Associates Publishers, London, England, 1991.