[International Workshop on Fundamental Research for the Future Generation of Natural Language Processing (FGNLP), July 1992, Manchester]

Linguistic Knowledge Acquisition From Corpora*

Jun-ichi TSUJII

Sofia ANANIADOU **Iris ARAD** **Satoshi SEKINE**

Centre for Computational Linguistics

University of Manchester Institute of Science and Technology

P.O.Box 88, Manchester M60 1QD U.K.

tsujii, effie, iris, sekine@ccl.umist.ac.uk

July 23, 1992

1 Introduction

After the intensive studies of grammar formalisms during the eighties, we are now witnessing the emergence of new research streams. The various names by which they are called, such as corpus-based linguistics, sublanguage-based NLP, example-based MT, statistic based NLP, etc., reflect the different techniques they use, the different research objectives they have, and their different conceptions about problems we encounter in the fields of computational linguistics and natural language processing. However, despite the differences, researchers in these fields also share common convictions such as:

1. The studies of grammar formalisms have been concerned with forms of linguistic knowledge. For example, grammar formalisms determine forms of lexical descriptions but they are not concerned with actual descriptions of individual words.

2. The studies of grammar formalisms have heavily relied on human intuition without any concrete evidence. It is often the case that constructions judged as intuitively ungrammatical appear in actual texts. In order to develop realistic descriptions of language, we have to observe language usages in actual texts.

3. The studies of grammar formalisms tend to treat only a restricted set of linguistic phenomena which theoretical linguists are interested in. We have to treat much wider sets of phenomena.

4. The studies of grammar formalisms have emphasised universality and generality as guiding principles in their research. However, actual language usages are full of idiosyncracies or specificities. We need paradigms by which we can treat these naturally, not as exceptions.

* We are grateful to Kutluk Ozguven for his contribution to discussions on the content of this paper, and to Maeda Toshiyuki and Patrick Olivier for their help in its preparation

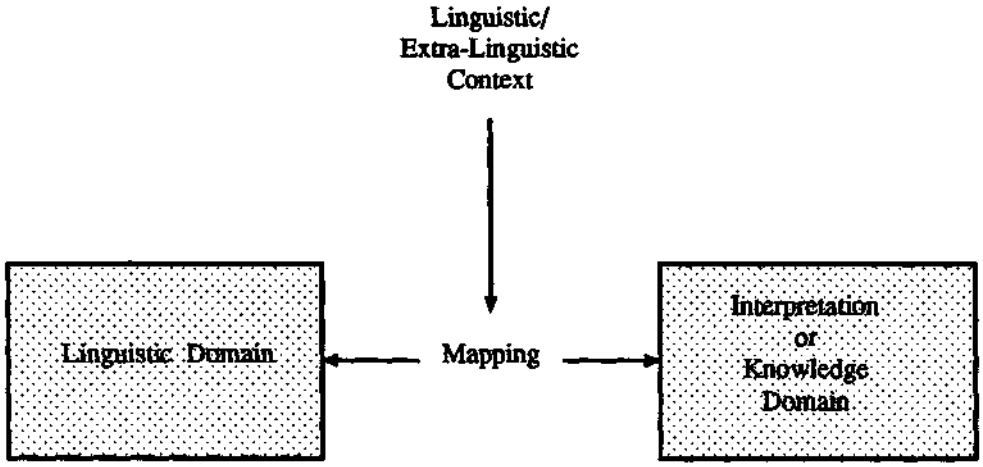

Figure 1: Schematic View of NLP from the Knowledge-based Perspective

Our research falls within the field of Sublanguage-based NLP, consequently we emphasise the importance of Knowledge Acquisition from Corpora and design methodologies for NLP systems specific to given *sublanguages* and *applications.* We also describe knowledge acquisition tools that we have developed and report on the results of experiments obtained using these tools.

2 Sublanguage-based NLP

2.1 Sublanguage-based NLP vs. Knowledge-based NLP

Most of serious difficulties we have encountered in natural language processing and its application seem to be consequences of the following two essential properties of natural language.

1. Context-Dependency of Interpretation of Natural Language Expressions: the same linguistic expressions have to be interpreted differently, depending on the context in which the expressions appear. In the case of machine translation, for example, the same expressions have to be translated differently.

2. Reliance of NL communication on Extra-linguistic Context: because human communication by language heavily relies on shared knowledge and shared environments between speakers (writers) and hearers (readers), the *linguistic* context itself often lacks the information necessary for determining how expressions in it should be interpreted.

This dependency of NL interpretation on context, especially on extra-linguistic context, has been addressed by two different camps, Knowledge-based NLP and Sublanguage-based NLP. Though they share a lot of common conceptions about the problems, they emphasise different aspects.

The differences between these two camps can be explained using the schematic view of NLP given in figure 1, which is more or less, an agreed view among researchers in Knowledge-based NLP.

In this scheme, there are two domains which are linked by a mapping: the *Linguistic Domain* and the *Interpretation Domain* (or *Knowledge Domain).* The *Linguistic Domain* consists of all possible expressions in language, while the *Knowledge Domain* is a domain in which interpretation results of expressions in the Linguistic Domain are represented. We do

not need to commit ourselves to specific internal organisations at this schematic level, though we can assume, for example, that the Linguistic Domain is defined by a set of generative rules (as the generative linguists did). We can also assume that specification of the Linguistic Domain consists of several levels of descriptions (as LFG does) and that the mapping between the two domains can refer to these levels of description.

Both camps, the Knowledge-based NLP camp and Sublanguage-based NLP camp, agree that the mapping between these two domains is context-dependent in the sense that the same entities in the Linguistic Domain are to be associated with different entities in the Knowledge Domain, depending on context.

The Knowledge-based NLP camp has claimed that, in order to capture the context-dependency of the mapping, we have to first of all explicitly represent (inside computer programs) all sorts of context, and that NLP systems have to be able to manipulate such explicitly expressed context to determine the interpretations of input sentences. They have tried to represent extra-linguistic contexts relevant to language interpretation, from deictic contexts such as the Blocks World in SHRDLU, to knowledge about a speaker's goals, or a hearer's knowledge about a speaker's knowledge, etc. All are implicit in linguistic expressions but play, so they claim, essential roles in language interpretation.

They have tried to show, by using explicitly represented contexts, what sorts of mechanisms are necessary to relate utterances, for example, with a speaker's goals which are implicit in linguistic expressions. They have also been interested how the contexts represented changes during the development of conversations and texts. In short, they are interested in what we call *local* context, and *dynamic* aspects of interaction between linguistic expressions and local context.

Their interest in dynamic aspects of context-dependency is well illustrated by their strong interest in problems related with anaphoric expressions. Anaphoric expressions should be interpreted differently even in the same texts or within the span of a conversation, because local contexts relevant to their interpretation change from one occurrence to another. Less obviously this is also the case for the interpretation of speaker intention. The structure of plans which speakers have may change in the due course of a conversation and lead to different interpretations of the same linguistic forms.

On the other hand, the Sublanguage-based NLP camp (which we belong to) see the context-dependency of language from a different perspective, or to be more precise, emphasise the influence of a different type of context, that is the *global* context in which texts are prepared or conversations take place. The type of context that we are interested in does not change according to the development of texts or conversations, but instead is established by *communicative environments*, eg. types of writers (specialists such as engineers, doctors, businessman, etc. or non-specialists), types of readers, levels of formality, subject domains of topics (business, technical fields such as computer technology, economics, etc.), etc.

Our view of Sublanguage-based NLP can be best illustrated by the schematic view in figure 2.

The whole scheme of Knowledge-based NLP is embedded in the above figure and thus all the components (Linguistic Domain, Knowledge Domain and the Mapping) are taken to be dependent on a more global context. It is obvious that the Knowledge Domain is highly dependent on subject domains which are established by global context. Though less obvious, the Linguistic Domain, which is often considered to be the same across different communicative environments, also varies from one sublanguage to another. There are, for example, expressions which appear only in specific sublanguages, that are generally taken to be ungrammatical. Or certain linguistic constructions which only rarely, occur in a given sublanguage.

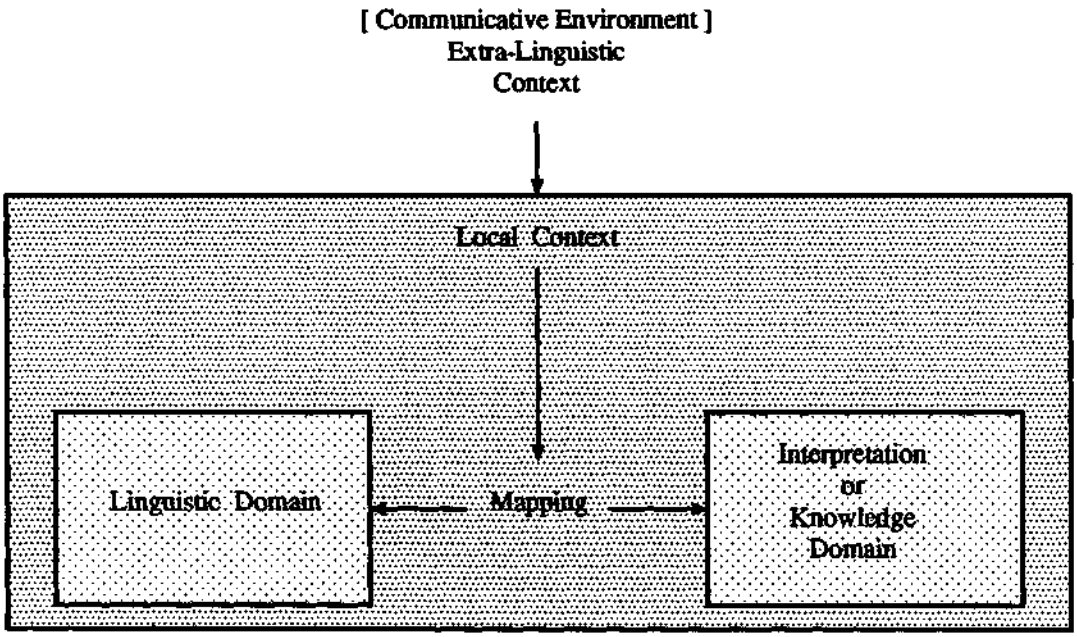

Figure 2: Schematic View of NLP from the Sublanguage-based Perspective

More importantly, there axe certain communicative environments in which language usage is, deliberately or accidentally, well regulated and well circumscribed. Therefore, the scheme prescribed by Knowledge-based NLP can be largely simplified.

The language used in meteorological reports which the MT system METEO treats, is an extreme example. It appears that the language looses some of the properties of human language, *creativity* and *infiniteness,* which make computer processing of language very difficult. The Linguistic Domain (the set of sentences), for example, is no longer infinite but is instead a finite set. As far as the language in this specific communicative environment is concerned, the mapping which links the two domains is rather straightforward and it scarcely shows the dynamic context dependency which Knowledge-based NLP camp has been concerned with. In other words, the language used in meteorological reports is a fairly impoverished version of general language, and follows restrictions imposed by the particular communicative environment.

The claim of Sublanguage-based NLP is:

1. When we confine ourselves to the processing of texts of particular types in particular subject domains, we can find a lot of restrictions or regularities which the language in that particular communicative environment follows.

2. Such restrictions lead to an impoverishment in the flexibility of language, which human language otherwise has.

3. Effective utilisation of restrictions imposed by global context would have more direct implications on performances of NLP systems than treatment of dynamic context dependency.

4. Not only performances but also architectures of NLP systems can be largely simplified in cases where the language to be treated looses its creativeness and infiniteness.

5. Because global context affects not only the Knowledge Domain but also Linguistic Domain, it is often the case that even linguistic knowledge like syntactic rules, parts-of-speech, etc. has to be changed according to the sublanguage.

2.2 Knowledge Acquisition and Scale-Up Problems

Knowledge-based NLP designers implicitly use restrictions imposed by global context, when they create their knowledge bases for specific subject domains and relate them with linguistic knowledge in lexicons.

The word "block", for example, appears only as a noun with a single meaning in Winograd's Blocks World, while it can be a verb. Even a small dictionary such as "Oxford Advanced Learner's Dictionary of Current English" (OALD) enumerates twelve different senses for the noun usage of "block".

It is certain that Winograd, though undeliberately, took into account such restrictions on the expressive power of the language in his domain, when he designed his program. If such restrictions had not taken into account and all possible senses of "to put", "to get", "block" etc. had been put into the lexicons, then his program would not have been able to exhibit such remarkable performance.

Similar restrictions can easily be found in less trivial and more realistic application environments. 114 out of 125 occurrences of the verb "to match" in the UNIX manual are translated into the same Japanese verb "icchisuru-suru", while a small-sized English-Japanese dictionary (published by Iwanami) lists fifteen different Japanese verbs as translation equivalents, among which "icchi-suru" is treated as one of less frequent translation equivalents.

The rest of the occurrences of "to match" in UNIX Manuals are translated as "taiou-suru" in Japanese. So we only have two different translations of "to match" in this sublanguage, despite the fact that there are 15 or more translations listed in a small dictionary. Thus restricted correspondences, similar to the types of restrictions which we saw in the Blocks World, exist in the sublanguage of UNIX Manuals.

Knowledge-based NLP research so far tends to take such restrictions imposed by global contexts for granted, and focuses on the dynamic interaction of interpretation with the local contexts which they were originally interested in. However, it is this ignoring of problems related with global context effects that results in the difficulties called *Scale-up Problems.* These problems are encountered whenever we try to apply a proto-type knowledge-based system to more realistic applications.

In a proto-type system, we usually assume:

1. The Knowledge Domain exists independently from language.

2. Actual content of the knowledge-base can easily be constructed, for example, by domain specialists.

3. Once the actual Knowledge Domain is constructed, the mapping between the Knowledge Domain and the Linguistic Domain can easily be defined, due to the fact that the mapping in specific domains is rather straightforward.

However, these assumptions have often proved to be wrong, except for few cases such as METEO where the complexities of sublanguages and their knowledge domains are so few that designers can capture the structure of the Knowledge Domains and the mutual relationship between the Linguistic Domain and the Knowledge Domain, through their own introspection.

The situation is not so simple in most realistic applications. Knowledge of computer technology, for example, which constitutes the knowledge domain for the translation of computer manuals, would be too vast for designers of NLP to capture through introspection.

First of all, knowledge of computer technology can mean anything from a shallow layman's view, through to that which computer scientists may have. It is not at all clear what

levels of knowledge are actually relevant to linguistic processing. Appropriate levels in the Knowledge Domain may depend on the tasks which application systems are supposed to perform (translation, abstraction, conversation, etc.), which is beyond the scope of this paper. But even if we confined the discussion to linguistic processing like the disambiguation of syntactic structures, it is not at all clear what level of knowledge is necessary for such processing.

This is partly due to the intricate mutual dependency between language and our knowledge of the domain. In actual application environments, we cannot determine the structures and the content of knowledge domains independently from linguistic domain. Revealing the structure of the Knowledge Domains, especially structures which are relevant to linguistic processing, can only be possible through the examination of actual texts in the subject domains.

Secondly, even though the mapping between the linguistic domain and the knowledge domain is straightforward in a given sublanguage, it is very difficult, if not impossible, to know the actual form of the mapping in advance. Our introspection about the usage of "to match" in the above is triggered by a discovery obtained by examining the actual corpus.

One of the claims of Sublanguage-based NLP which we would like to emphasise in this paper is the importance of *Knowledge Acquisition* from a corpus such as the above. By emphasising the influence of (or restrictions imposed by) global context on the Linguistic Domain and the form of mapping for interpretation, the Sublanguage-based NLP camp commits itself to the discovery of such influences and restrictions. In particular, to the process of discovery through corpus study, which the Knowledge-based NLP camp has largely ignored.

2.3 Word Senses vs. Denotations

Though we would do not want to discuss philosophical problems related with senses and denotations, we would like to make some brief comments to clarify the differences between the two camps.

1. The Knowledge-based NLP camp has only one component responsible for treating context-dependency. As a result, they tend to treat all sorts of context-dependency inside a single system. This means that they tend to be interested in "senses" of words which can be used to determine specific "denotations" in various different Knowledge Domains. They sometimes talk, for example, about concepts expressed by words, which are close to the "senses" of words and which are universal in that they are independent from individual Knowledge Domains.

2. The sublanguage-based NLP camp tends to be more pragmatic. If the verb "to match" is used only to express very specific states or actions in UNIX manuals such as "a left parenthesis matches a right-parenthesis", "a string of characters matches another string of characters", they tend to relate the verb directly with these two *denotations* in the Knowledge Domain, even though these two usages in UNIX manuals are very specific realisations of particular senses of the word.

3 Corpus-based Linguistics vs. Sublanguage-based NLP

3.1 Sublanguage-based NLP as an Engineering Practice

The emphasis on the influence of global context, and the pragmatic attitude discussed in the above also distinguishes Sublanguage-based research from general Corpus-based Linguistics. While corpus-based linguistics tends to be concerned with comprehensive descriptions of language and making linguistically meaningful claims, Sublanguage-based NLP is not so interested in making general claims based on results obtained from corpus study. The general claim we would like to make is about the effectiveness of the methodology or procedures for extracting knowledge from a given sublanguage corpus.

Linguists or lexicographers involved in corpus-based linguistics, for example, may be interested in accumulating as many usages of specific words as possible, and then by examining them with help of statistical programs or others, they try to establish different *senses* of the words, which are abstractions of usages across different Knowledge Domains and are valid independently of an individual Knowledge Domain. The results thus obtained (eg. knowledge extracted from a corpus) are general claims as the meanings of the specific words, and are in themselves linguistically valuable claims.

On the other hand, in Sublanguage-based NLP, we are not interested in accumulating comprehensive usages of words across different domains, nor in discovering a list of word senses. Instead, we are interested in discovering specificity of word usages in given sublanguages, and how much restriction the global context of a given sublanguage imposes on the interpretation of given words.

The discovery that the verb "to match" is used in UNIX manuals to describe one or two very specific situations does not lead to any valuable linguistic claim, but may have significant implications in the design of a NLP system to treat computer manuals.

Our claims concerning Knowledge Acquisition in Sublanguage-based NLP are:

1. The enumeration and formulation of word *senses* requires the ability to abstract common properties, which real-world entities (or concepts) described by the words share. Such an ability assumes huge amounts of extra-linguistic knowledge about different Knowledge Domains (for recognising *denotations),* and some intuition by which common properties of denotations in different knowledge domains are recognised.

2. We cannot expect computer systems at present to have such vast knowledge across different domains and intuition for abstracting common properties of different denotations in different Knowledge Domains.

3. More importantly, in a mixed corpus containing different sublanguages every possible word in the corpus may denote every possible denotation in different Knowledge Domains. Meaningful discrimination of *senses* cannot be obtained by any automatic or semi-automatic means.

4. It is more sensible and plausible to think of an automatic or semi-automatic procedure which discovers *denotations* of words in specific sublanguages, and Knowledge Domain structures of the sublanguages simultaneously.

We agree with the following claim made by [Calzolari and Bindi, 1990]:

> "An important prior concern which strongly influences the quality of the results is the overall design of the corpus... Different selections of texts are in fact

necessary according to the type of task which is to be carried out... As far as lexical/semantic data are concerned, for example the extraction of compounds, especially those belonging to a specific sublanguage, is best accomplished when working on a non-balanced corpus (some results could be flattened in balanced corpus), but on a specific corpus for the sublanguage."

3.2 The Corpus in Sublanguage-based NLP

As any other engineering practice, we are constrained by requirements of practical application environments.

Some statistics-based procedures which could potentially produce useful results are not usable for knowledge acquisition in Sublanguage-based NLP, simply because they require a huge corpus. We assume the availability of a certain amount of computer-usable corpora in given sublanguages, but statistics-based procedures often require far more data to get "reasonable" results. While structurally tagged corpora are generally more useful than non-tagged corpora to train a system's knowledge or to discover effective regularities in sublanguages, the discovery procedures based on them are unlikely to be viable for the same performance-based reasons. To prepare from scratch, a large structurally tagged corpus in a given sublanguage is not cost-effective. To be more precise, we need either (1) an automatic or semi-automatic procedure to reduce the burden of tagging corpora structurally, or (2) procedures which produce tagged corpora and perform discoveries at the same time (see 5.2).

4 Knowledge Acquisition Systems

4.1 Object-Systems vs. Meta-Systems

Recent inclination towards Corpus-based linguistics or Sublanguage-based NLP can be seen as a reaction against the research of the eighties, which is largely characterised by its fondness for formalism studies. Formalism studies tend to ignore actual linguistic knowledge which should be described by formalisms.

Formalism research also emphasises the "universality" or "generality" of proposed formalisms so that linguistic phenomena which are not considered universal, such as sublanguage-specific phenomena, have largely been ignored, in a similar fashion to the Chomskian school which classifies them as "performance-related".

From the view point of proposed architectures for NLP systems, the reaction against the eighties can be categorised as follows:

1. **Total Rejection//** Connectionist NLP [McLean, 1992], Example-based MT [Jones, 1992] or NLP, Statistics-based MT [Brown et al., 1990].
   This camp most strongly most strongly systems based on formalism studies and proposes system architectures which do not use any linguistic concept whose existence is a priori assumed in grammar formalisms (eg. constituent structures, semantic representation of some sorts, a set of "rules" based on such representations which predicts a set of possible linguistic expressions or which specifies relationships between different levels of representations, etc.).

2. **Add-On**
   Knowledge-Acquisition or Knowledge-Generation Component as a Meta-Component.

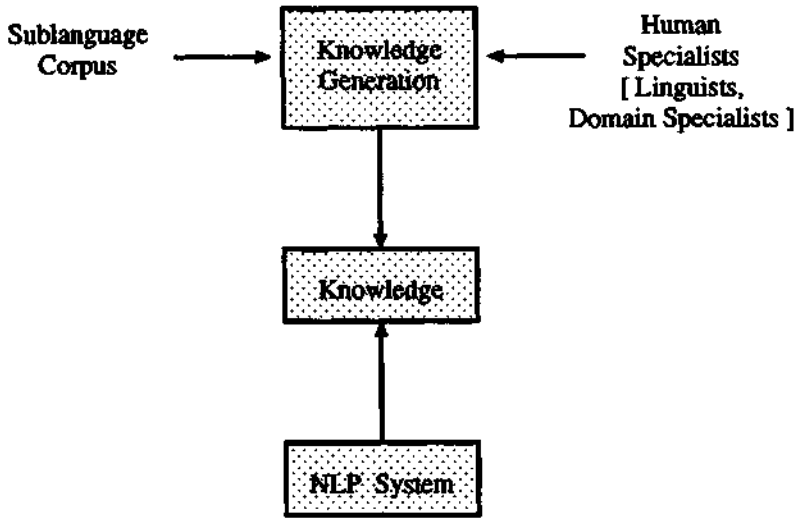

Figure 3: Knowledge Acquisition as an Add-on Component

This group, which we belong to, assumes rather conventional, rule-based NLP systems and is interested in an add-on component for producing actual knowledge or "rules" which are then used by conventional NLP systems (object-level systems) (see figure 3). The knowledge acquisition component produces, for example, a set of new syntactic rules which treat constructions specific to a sublanguage, or produces a set of semantic classes which are effective only in a given sublanguage for specifying selectional restrictions.

3. Parasitic

Tuning of an existing set of rules.

This group assumes a fixed set of knowledge and tries to add to them extra information derived from a corpus, such as relative frequencies of individual rules. Such statistical information is used as preferential cues in choosing plausible parsing results. There has already been quite a lot research showing that such statistical knowledge can contribute significantly to the selection of correct syntactic parses [Garside and Leech, 1985].

Each of these three approaches attempts to use corpora in their own way to formulate actual "knowledge", which formalism studies have largely ignored. **Total rejection** approaches reject even the representational forms of knowledge suggested by formalism studies, and hence also rejects processing mechanisms based on them. They devise their own processing mechanisms based on completely different representational forms of knowledge such as connection networks, example storage (corpus), etc.

On the other hand, **add-on** approaches accept the forms or types of knowledge proposed by studies of formalisms and tries to produce instances of knowledge which reflect actual usages of language occurring in the corpus. Because the **add-on** approach inherits the forms of knowledge and forms of linguistic descriptions from conventional studies, the architecture of conventional NLP systems can be assumed as the object level NLP system.

In the **parasitic** approach, performance shown in the corpus is only reflected in a form of statistical data attached to rules. The "knowledge" extracted from the corpus can modify the system behaviour, but there may be severe restrictions on possible modification. We cannot expect systems to accept constructions which are taken as ungrammatical by the original rule set, nor can we expect systems to impose semantic constraints specific to a given sublanguage.

We adopt the **add-on** approach as our research paradigm for Sublanguage-based NLP. The following is a list of the main reasons for our choice, though we cannot discuss them fully in this paper:

1. **Amount of Data Required**
   As we have seen in 3.2, insisting on a large sublanguage corpus is prohibitive in most application environments. This restriction make the choice of the **total rejection** approach less reasonable. Because the **total rejection** approach rejects any knowledge speculated by linguists, its systems have to re-discover all linguistic regularities from a sample corpus that is rather restricted in size.

2. **Transparency and Controllability of System Behaviour**
   For the same reason as above, we think that human intervention is inevitable at certain stages in knowledge formulation. This rules out the possibility of using paradigms, such as connectionism or purely statistics based paradigms, whose internal knowledge representations are not accessible to human designers.

3. **Importance of Linguistic Structures**
   It is our conviction that most NLP application systems have to manipulate some type of structural description for sentences (or texts), eg. transforming them, translating them to structural representations of other sorts, etc. Therefore, the utility of frameworks which do not or cannot represent *structures* explicitly is severely reduced.

4. **Specificities of Sublanguages**
   In order to treat the specificities of sublanguage discussed so far, in the **parasitic** approach, mere tuning of behaviours of object systems by statistical measures, may not be sufficient in most cases. In particular, the mapping between the Linguistic Domain and the Knowledge Domain, and the internal organisation of the Knowledge Domain, are highly dependent on individual sublanguages. Therefore we have to create new units of knowledge, which involves more than just modifying existing knowledge.

4.2 The Knowledge Acquisition Component

The similarity between the schematic view of Sublanguage-based NLP in figure 2 and the construction shown in figure 3 is obvious. The global context or communicative environment in figure 2 is replaced by the *Knowledge Generator* in figure 3. This illustrates the role of the *Knowledge Generator* or *Knowledge Acquisition Component* in the whole construction of Sublanguage-based NLP systems.

The following points characterise the Knowledge Acquisition (KA) Component:

1. The KA Component is responsible for treating the influence of global contexts. Unlike Knowledge-based NLP, we are interested in the *static* (not *dynamic)* influence of *global* context so that the KA Component needs not to be invoked dynamically during the actual processing of object-level NLP systems.

2. The KA Component coordinates the two main knowledge sources, the sample corpus and human intuition. Partly due to practical reasons (eg. the size of the corpus usually available in a sublanguage) and partly as a result of our theoretical position (eg. context, especially *global* context, is completely implicit in texts), we believe that human intuition or introspection is indispensable in the analysis of the effect of global context on language usage.

3. The KA Component assumes the forms of knowledge on which the processing mechanisms of object-level NLP systems depend. In other words, the forms of knowledge are imposed by object-level NLP systems. Further, the forms imposed by NLP systems are more or less the same as formalism studies propose. We assume, for example, the existence of subcategorization frames for individual predicates around which semantic constraints will be specified. Though we tentatively take this position, it may be the case that the forms of knowledge themselves have to be changed for some classes of sublanguages. Syntactic knowledge for very restricted sublanguages like the language in METEO, for example, may not require a rule system which potentially defines an infinite set of expressions.

4. The KA component consists of a set of software tools which share common data bases. Though quite a few procedures based on statistical methods have been proposed, none of them can automatically produce legitimate knowledge or sufficient cues for humans intervention, for example, to determine the content of the Knowledge Domain. As [Bindi et al., 1991] claims, we have to have a set of tools which collectively provide sufficient cues to guide human introspection in the appropriate direction. In order to integrate various types of tools which may work on different levels of descriptions of the same corpus, the KA Component should have an integrated data base management system which maintains the mutual relationships among corpus descriptions at different levels, knowledge hypothesised by programs or humans, etc.

A detailed explanation of the software in KA Component and the data base organization is given in [Tsujii et al., 1991].

5 Individual Tools

Tools for knowledge acquisition which we have developed, include programs proposed and used by other groups, such as those for *Mutual Information* and *Clustering.* They also include standard tools such as parsers of various kinds, a graphical debugger for parsers, etc. In this section, we will give brief descriptions of the tools which our group has developed, and we will illustrate how they can be used. More detailed descriptions of these tools can be found in [Arad, 1991], [Sekine et al., 1992a] and [Sekine et al., 1992b].

5.1 From Contexts to Classifications of Words

5.1.1 Inversion of KWIC

One of the basic assumptions in corpus-based linguistics is that linguistic contexts have enough information to characterise properties of words (or phrases) or to get appropriate classification of words. This assumption is the foundation for developing automatic procedures which classify words in terms of the linguistic contexts in which they occur. Such procedures have to perform the following two tasks:

1. Discovery of effective ways of characterising linguistic contexts.

2. Discovery of word classes based on contexts characterised in 1.

Though we can think of statistical procedures which could perform these two discovery processes simultaneously (from scratch), such procedures surely require a large amount of

data. In particular, semantic classifications which are specific to individual sublanguages and therefore have to be discovered from scratch require a sophisticated mechanism for characterising contexts. Discovering an effective way of characterising contexts from scratch (and from a relatively small non-tagged corpus) appears to be very difficult, if not impossible.

The other extreme is to classify words by human inspection. While we will discuss statistical procedures in 5.2, we focus here on human classification and a tool we developed to aid the process.

One of the conventional tools for aiding human inspection of a corpus is a concordance program (KWIC). This tool, while useful for inspecting a set of contexts where a specific word occurs, is not so successful in helping humans to to inspect a set of contexts where a specific word occurs.

While KWIC lists a set of contexts sharing the same word, and displays them with the shared word as an index, the tool (CIWK) we developed does the opposite [Arad, 1991]. CIWK gathers a set of words sharing the same context[1] and shows them with the shared context. CIWK shows groups of words which occur in the context < a_1, a_2..., a_n, *, b_1, b_2.., b_m >, where * is the position of the word belonging to a group and a_i and b_j are the words which constitute a context, n and m are given as parameters to CIWK by the user.

5.1.2 Results Produced by CIWK

The data shown by CIWK is rather unexpectedly interesting, enough so to merit human introspection. The data can be used in a number of different ways including automatic phrase recognition, discovery of paradigmatic relations etc., the following discussion concentrates on its use in semantic classification. As expected, stricter contexts such as [4,3], [4,2] ([4,3] means that n and m are 4 and 3, respectively) produce a set of groups whose members stand in semantically close relations (such as synonyms and antonyms). The following are examples of groups produced by [4,1].

The first two groups of the following are antonyms which hold in general language. The third is also a antonym group in a broader sense, but this may only be the case in the restricted domain of UNIX Manuals.

```
*************************
leftmost
rightmost
 GROUP: 14       FREQUENCY: 2
 KEY: RE to match the * portion
*************************
next
previous
 GROUP: 17       FREQUENCY: 2
 KEY: Skip to the ith * filename
*************************
```

[1] Context or environment is defined in terms of words immediately preceding and following the index term. While concordance programs are not always feasible for defining the immediate environment of the index term, taking the entire sentence as the "context", our tool allows the user to specify and/or modify the required context in terms of the number of preceding and following words. Varying the environment allows the user to inspect different classification results.

```
*************************
existing
new
 GROUP:   160   FREQUENCY: 2
 KEY:  only be made to    *    files
*************************
```

As for "new" and "existing" in the above, we can find many groups whose members have very close semantic relations in this specific domain but are only remotely related in general language. That is, there are groups of words whose denotations in this given Knowledge Domain, are closely related, but whose senses axe only remotely related, if at all. The following are examples:

```
*************************
given
specified
 GROUP:  5     FREQUENCY:  2
 KEY:  If no filename is * the
*************************
given
omitted
 GROUP:  7     FREQUENCY:  2
 KEY:  If this option is * sort
*************************
*************************
described
discussed
listed
 GROUP:   204  FREQUENCY: 3
 KEY:  the generic tool arguments * in
*************************
*************************
caught
ignored
 GROUP:  36    FREQUENCY:  2
 KEY:  all signals currently being  * or
*************************
```

The pair in GROUP:36, "to catch" and "to ignore" can hardly be thought of as antonyms in general language. On the other hand, the actions denoted in the context of GROUP:36 clearly constitute contrastive pairs (of actions) in the Knowledge Domain and the two words are mutually replaceable in this linguistic context to express either of the two contrastive actions.

Actually, examining KWIC for these two words reveals that all occurrences of the word "to catch" (6 occurrences: 3 in active voice and 3 in passive voice) are in similar linguistic contexts where the object to be caught is either "signal" (or "signals") or INTERRUPT. The

verb can be replaced in all these contexts. On the other hand, "to ignore" occurs far more often and there are many occurrences which cannot be replaced by "to catch" such as:

```
   This option is   ignored    if the terminal does not have the
Comments are also   ignored  ,  except that a comment terminates
classification are  ignored  .
```

Thus, by examining CIWK and KWIC and using our introspection, we can see that: (1) there is a pair of contrastive actions in this Knowledge Domain, which is expressed by "to catch" and "to ignore"; (2) the occurrences of "to catch" in this corpus always denote one of the contrastive actions; (3) "to ignore" has other usages which denote actions other than the one in this contrastive pair; (4) the actions in this pair take entities denoted by "signal", INTERRUPT, etc.

"To ignore" in the corpus may be used with the same sense, but because of the above considerations, it may be reasonable to establish separate denotations (one of which is the one contrastive with "to catch") in order to specify constraints at the knowledge level.

5.2 Description of the Corpus and Semantic Classification

5.2.1 Gradual Approximation

We discovered in 5.1.2, that through inspection of the corpus using KWIC and CIWK, and also through our introspection regarding the Knowledge Domain, that:

"SIGNAL-like-words appear as deep-object of CATCH-like-actions"

In this process, we mentally transform surface sentences into standard forms of a certain level (for example, abstract-syntax) by reverting passive voice into active voice, or by omitting adverbs, adjectives, etc. which often intervene between the verbs (to ignore, "to catch") and the nouns which are head-nouns of noun phrases occupying the object-position.

It seems reasonable to assume that information concerning collocations between verbs and nouns such as "a certain noun often appears as deep-subject of a certain verb" gives effective linguistic contexts for semantic classification of words.

We can of course, imagine a fully automatic statistical procedure which discovers (from an un-tagged corpus, without any a prior linguistic knowledge) both effective linguistic contexts for semantic classification and semantic classes simultaneously. But then, this procedure may have to (possibly implicitly) discover:

1. *basic linguistic concepts:* it has to discover not only structural concepts (such as noun phrases, parts-of-speech, etc.) to grasp the structure of sentences in the corpus, but also inter-structural operations such as "surface-subject in passive voice plays the role of deep-object (the same role of surface-object in active voice)" etc.

2. *effective contexts for semantic classification:* it has to choose contexts such as governor-dependent relationships at the deep syntax level as effective contexts, amongst a vast number of other possible characterisations of linguistic contexts.

We cannot judge whether such discoveries are possible or not through pure statistic means, nor can we judge whether the procedure has to invent it own structural descriptions at a certain stage of processing. However, considering the sheer size of the corpus which

we believe such a procedure requires, it seems inapplicable for knowledge acquisition in Sublanguage-based NLP (see 3.2).

On the other hand, it is too demanding in most actual application environments to require a reasonable amount of structurally tagged corpus, though a corpus annotated with syntactic structures is necessary to avoid the above difficulties.

One possibility is to use an existing syntactic parser to tag the corpus, but it is a well-known fact that the determination of syntactic structures of sentences is not possible by syntactic knowledge alone, but requires semantic knowledge to prevent proliferation of possible syntactic structures. This is the very knowledge which we are trying to discover.

This is typical of 'chicken-and-egg' situations which knowledge acquisition programs encounter. For example programs need a properly tagged corpus to learn certain types of knowledge, but tagging a corpus properly requires not only other kinds of a prior linguistic knowledge (this requirement itself may be blamed as serious retreat by purists who tend to refuse all a prior linguistic knowledge) but also the knowledge to be learned.

The strategy we adopted to break the circularity was to use roughly approximated, imperfect knowledge of semantic domains in order to hypothesise correct syntactic structures for sentences in a corpus [Sekine et al., 1992a] [Sekine et al., 1992b]. Because such approximated semantic knowledge will contain errors or lack necessary information, syntactic structures assigned to sentences in a corpus may contain errors or imperfections.

However, if a program or human expert produces more accurate, less imperfect knowledge of the semantic domain from descriptions of the corpus (assigned syntactic structures), we can use this to produce more accurate, less erroneous syntactic descriptions. The same process can be repeated again to gain further *improvement* both in the knowledge of the semantic domain and in syntactic descriptions of the corpus. Thus, we may be able to converge gradually on both correct syntactic descriptions of a corpus, and semantic classifications of words.

The same idea, which we call *Gradual Approximation,* can be applied to other learning problems. *Gradual Approximation* works as follows:

1. Two types of data are kept: a tentative description of the corpus obtained by the current hypothesised linguistic knowledge, and the currently hypothesised knowledge.

2. The current corpus description is used by a human or computer programs to produce (or learn) better hypothesised knowledge.

3. The knowledge produced in 2. produces a better description of the corpus. Repeat the step 2. and 3. until the process converges.

5.2.2 Experiments

(a) Discovering Semantic Collocations

One program based on *Gradual Approximation* discovers semantic collocations between words. *Plausibility Values* for individual collocations are calculated, and are then used to choose preferred readings. The program is general in the sense that it can be applied to any syntactic construction which leads to ambiguous structural descriptions. In our experiments, it was applied to determine attachment positions of prepositional phrases (PP-attach), and also to determine the structure of Japanese compound nouns (which consist of sequences of two or more nouns).

The program first produces all possible syntactic descriptions of sentences in a corpus (the first approximation of a corpus description). Based on the description obtained, it proceeds to

compute the first approximations for plausibility values of individual collocations (which are basically frequencies of individual collocations found in the first approximation of the corpus description, though there are some sophistications). The first approximation of plausibility values are then used to produce the second approximation of the corpus description, which in turn gives rise to better approximations for the plausibility values, and so on. The following are the data and the results of simple experiments:

```
Data :

  Sentences:
      I saw a girl with a telescope.
      I saw a girl with a scarf.
      I saw a girl with a necklace.
      I saw the moon with a telescope.
      I meet a girl with a telescope.
      A girl with a scarf saw me.
      I saw a girl without a scarf.

  Semantic Distances:{This data is used to compensate
      lack of a large corpus, by counting occurrences of
      semantically related words as occurrences of the
      words themselves - See the following section [b]}

      0.2 = {with without}
      0.2 = {scarf necklace}
      0.3 = {saw meet}
      1.0 = between unspecified words
```

From this data, we get the following results at the end of the first cycle (Table 1). The numbers in this table are the plausibility values of hypothesis-tuples between the words in corresponding columns. The plausibility value of the hypothesis-tuple (saw, WITH, telescope), for example, is 0.75.

	telescope	scarf	necklace
saw WITH	0.75	0.50	0.50
girl WITH 0.75	1.00	0.50	
moon WITH	0.50		
meet WITH	0.50		
saw WITHOUT	-	0.50	
girl WITHOUT	-	0.50	

Table 1: Plausibility values after the first cycle

As the result of *Gradual Approximation,* we get the following results after the fifth cycle (Table 2). Compared with the plausibility values after the first cycle, we can see that the plausibility values in this table are considerably more polarised. The plausibility values of semantically possible collocational pairs are approaching 1, while the others fall off to zero.

	telescope	scarf	necklace
saw WITH	1.00	0.26	0.30
girl WITH	0.93	1.00	0.99
moon WITH	0.00	0.00	0.00
meet WITH	0.57	0.04	0.04
saw WITHOUT	0.64	0.01	0.01
girl WITHOUT	0.58	1.00	0.64

Table 2: Plausibility values after the fifth cycle

We also have applied the same program to determine structures of compound nouns in Japanese (the corpus consisted of 616 compound nouns). Table 3 shows the proportion of correct analyses.

Words	correct	incorrect	indefinite	uncertain
3	66	29	5	13
4	41	7	5	1
5	4	0	0	2
total	111	36	10	16
(%)	(70.7)	(22.9)	(6.4)	(-)

Table 3: Results of experiment with compound nouns

(b) Noun Semantic Classes

The second *Gradual Approximation* program contains the first program in its larger repetition cycle. The final output of the program is a set of semantic classes of nouns, which are computed by a clustering program based on the semantic collocations computed by the program in (a). That is, we basically assume that the collocation between two words, where one of them occupies the position of direct dependents of the other, provides important cues to classify words semantically. Therefore such characterisation serves as an effective characterisation of linguistic contexts for semantic classification.

Though the validity of this assumption remains to be proved, it seems reasonable. More or less similar assumptions have been adopted by research groups at New York University [Grishman et al., 1986] and AT&T Bell Laboratories [Church, 1988] and [Hindle and Rooth, 1991]. However, in order to follow this assumption, we have to recognise pairs of words which stand in direct governor-dependent relations in a corpus. In the case of [Grishman et al., 1986], such relations were recovered mostly through human intervention, whilst only a restricted relation of SUBJ-VERB-OBJ was recognised in the parsed results of the Fidditch-parser for the experiments at Bell Laboratories.

As a result of our desires to minimise human intervention and maximise the utility of the corpus, we attempted to use *Gradual Approximation* to obtain all possible governor-dependent relationships in the corpus.

The program described in (a) can produce, without any human intervention, the set of all possible structural descriptions of sentences, some of which may be wrong (eg. it contains all possible structures for syntactically ambiguous sentences). However, as we saw before the program (a) computes the plausibility values of individual collocational pairs,

and the structural descriptions of sentences which contain less plausible collocations have lower plausibility values. So we can use these structural descriptions together with their plausibility values as contexts for semantic clustering.

In addition, the results of clustering are in turn used to accelerate the convergence of program (a) by using the *Similar Hypothesis Effect* [Sekine et al., 1992b].

The following two tables show the results of Japanese compound noun structure determination, after the first and second cycles. We can see that, though the improvement is not remarkable, a significant increase in the proportion of correctly recognised structures is achieved.

Words	correct	incorrect	indefinite	uncertain
3	72	33	4	14
4	41	7	5	1
5	4	0	0	2
total	117	40	9	16
(%)	(70.5)	(24.1)	(5.4)	(-)

Table 4: Results of experiment (b) after first cycle

Words	correct	incorrect	indefinite	uncertain
3	76	30	3	14
4	43	5	5	1
5	4	0	0	2
total	123	35	8	16
(%)	(74.1)	(21.1)	(4.8)	(-)

Table 5: Results of experiment (b) after second cycle

5.3 Adaptation of Existing Knowledge

The tools in the preceding two sections are mainly concerned with discovering semantic classes in the Knowledge Domain, which have to be formulated for individual sublanguages. The other claim in Sublanguage-based NLP is that even the Linguistic Domain is dependent on global context, so we have to revise existing knowledge or create new rules. The process required for this is similar to what we usually refer to as the grammar rule "debugging", thus tools such as rule application tracers, parse result visualisers, etc. can be used. However, while the debugging process in a conventional sense is taken to be the process of changing existing knowledge (or grammar rules) in order for them to reflect "legitimate" regularities of of human language, the debugging in Sublanguage-based NLP is the process of changing the existing knowledge to reflect regularities implicit in a sublanguage corpus.

Using this new conception of debugging, we can propose a combination of Corpus-based tools with conventional debuggers to systematise the process. For example, we can easily identify words whose lexical descriptions are wrong, using indices obtained from the following formula. The indices show the frequency of sentences containing a word that cannot be parsed by the current grammar.

$$FR(word) = \frac{\textit{(number } \textbf{\textit{of fault—parsed sentences in}} \textit{ which the word occurred)}}{\textit{(number of sentences in which the word occurred)}} = \frac{\textbf{\textit{(number}} \textit{ of fault—}\textbf{\textit{parsed sentences)}} \textit{ /(number of sentences)}}{}$$

defaults	1	10	1.298346
performed	1	10	1.298346
dependency	1	10	1.298346
item	2	19	1.292163
write	2	18	1.285363
so	1	9	1.285363
tset	1	9	1.285363
panel	1	9	1.285363

Table 6: Indices calculated for UNIX Manual Corpus

Table 6 shows the indices computed for the current grammar and the UNIX Manual corpus. The words with high index values are likely to have erroneous lexical descriptions, or lack necessary descriptions. Using the indices and KWIC, we can easily locate the failure causes such as: the lexical description of "write" only contains the verb description, so that frequent phrases like "the *write* command" cannot be parsed properly. Note that this type of conversion from a verb to a noun is one of the most frequently observed deviations for sublanguages from general language.

We are now extending this idea further to detect patterns of words or parts-of-speech which are likely to be the cause of parsing failures.

6 Concluding Remarks

The points we want to make in this paper are:

1. Serious difficulties in NLP are caused by the context-dependency of language interpretation.

2. Context which affects interpretation of linguistic expressions can be classified into two types, *local* context and *global* context (or *communicative environments).* While Knowledge-based NLP is mainly concerned with the *dynamic* interaction of language interpretation with local context, Sublanguage-based NLP is concerned with the *static* interaction of language interpretation with the global context.

3. A language used in a specific global context is called a *sublanguage.*

4. There are certain communicative environments where language usages are, deliberately or undeliberately, regulated and well circumscribed. Sublanguages in such global context can be processed easily by computer.

5. However, even though certain sublanguages loose their creativity and infiniteness, and show rather strict regularities, to discover such regularities with human intuition alone is not easy. We need methodologies by which we can discover them systematically.

6. Techniques similar to those developed in Corpus-based linguistics can be used to discover such regularities.

7. Though Sublanguage-based NLP research shares common techniques with Corpus-based linguistics, Sublanguage-based researchers are not so concerned about making linguistically valuable statements. We are more concerned with design methodologies for NLP systems and methodologies by which we can reveal the structure of Knowledge Domains for given sublanguages.

8. We chose the Meta-system approach in which Knowledge Acquisition Component is assumed independently from the object-level NLP system. The Knowledge Acquisition Component embodies the methodologies of the above in the form of a set of software tools.

9. Unlike Knowledge-based NLP systems, problems related with the context dependency of language interpretation are treated by the Knowledge Acquisition Component during the design phase of NLP systems. We expect the architectures of object-level NLP systems to be much simpler than in Knowledge-based NLP systems.

10. We illustrated some of the tools we developed, which are good examples of tools concerned with different aspects of Knowledge Acquisition in Sublanguage-based NLP.

11. CIWK is useful human introspective trigger for discovering the structure of a Knowledge Domain, and the mapping between the two domains (the Linguistic and Knowledge Domains).

12. The idea of Gradual Approximation in which linguistic knowledge and corpus descriptions are gradually simultaneously improved, can be applied to various types of problems.

13. Corpus-based techniques can be used to systematise the process of debugging grammar rules.

References

[Ananiadou, 1990] Sofia Ananiadou. Sublanguage Studies as the Basis for Computer Support for Multilingual Communication. *Proceedings of Termplan '90, Kuala Lumpur,* 1990

[Arad, 1991] Iris Arad. A Quasi-Statistical Approach to Automatic Generation of Linguistic Knowledge. *Ph.D Thesis, UMIST, Manchester,* 1991

[Bindi et al., 1991] R.Bindi, Nicoletta Calzolari, M.Monachini and Vito Pirrelli. Lexical Knowledge Acquisition form Textual Corpora: A Multivariate Statistic Approach as an Integration to Traditional Methodologies. *Instituto di Linguistica Computazionale del C.N.R. Pisa, Dipartimento di Linguistica dell'Universita di Pisa, Italy,* 1991

[Brown et al., 1990] Peter Brown, John Cocke, Stephen A.Della Pietra, Vincent J.Della Pietra, Fredrick Jelinek, John D.Lafferty, Robert L.Mercer, Paul S.Roossin. A statistical approach to machine translation. *Computational Linguistics,* 16(2) 79-85, 1990

[Calzolari and Bindi, 1990] Nicoletta Calzolari and Remo Bindi. Acquisition of Lexical Information from a large Italian Corpus. *18th COLING-90,* 1990

[Church, 1988] Kenneth Ward Church. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics,* 16(1)22-29, March 1990.

[Furuse, 1992] Osamu Furuse and Hitoshi Iida. An Example-Based Method for Transfer-Driven Machine Translation. *TMI-92, Montreal, Canada,* 1992.

[Garside and Leech, 1985] Roger Garside and Fanny Leech. A Probabilistic Parser *2nd Conference of the European Chapter of the A.C.L.,* 1985.

[Grishman et al., 1986] Ralph Grishman, Lynette Hirschman and Ngo Thanh Nhan. Discovery Procedures for Sublanguage Selectional Patterns: Initial Experiments. *Comp. Linguistics Vol.12 No.3,* 1986

[Hindle and Rooth, 1991] Donald Hindle and Mats Rooth. Structural Ambiguity and Lexical Relations. *29th Conference of the A.C.L.,* 1991.

[Jones, 1992] Daniel Jones. Non-hybrid Example-Based Machine Translation Architectures. *TMI-92, Montreal, Canada,* 1992

[McLean, 1992] Ian J. McLean Example-Based Machine Translation using Connectionist Matching. *TMI-92, Montreal, Canada,* 1992

[Nagao, 1984] Makoto Nagao. Towards A Framework of a Mechanical Translation between Japanese and English by Analogy, in *Artificial Intelligence and Human Intelligence, (ed:A.Elithorn and R.Banerji, North-Holland,* 1984.

[Sato, 1990] Satoshi Sato and Makoto Nagao. Towards Memory-based Translation. *Coling 90, Helsinki, Finland,* 1990.

[Sekine et al., 1992a] S.Sekine, J.J.Carroll, S.Ananiadou and J. Tsujii. Automatic Learning for Semantic Collocation. *3rd Conference on ANLP, Trento, Italy,* 1992

[Sekine et al., 1992b] S.Sekine, S.Ananiadou, J.Carroll and J.Tsujii. Linguistic Knowledge Generator. *Coling 92, France,* 1992

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

[Zernik and Jacobs, 1990] Uri Zernik and Paul Jacobs. Tagging for Learning: Collecting thematic relations from Corpus. *13th COLING-90,* 1990