Reference: *Proceedings of the International Conference Recent Advances in Natural Language Processing, Tzigov Chark, Bulgaria, 12-15 September*

Learning Terminology in a Foreign Language

Vania Dimitrova
Department of Computer Science
Faculty of Math. and Informatics
Shoumen University, Bulgaria
e-mail: dim@uni-shoumen.bg

Darina Dicheva
Department of Computer Science
Faculty of Math. and Informatics
Sofia University, Bulgaria
e-mail: darinad@fmi.uni-sofia.bg

Abstract

This paper suggests an application oriented approach that concerns adopting of some NLP techniques in a vocabulary CALL for terminology learning. We discuss how the lexicon, word formation rules, and term semantics are used for diagnosing the learner's knowledge and guiding the instruction.

1 Introduction

Intelligent Computer-Assisted Language Learning (ICALL) systems are starting to emerge in the last couple of years as a separate field in the Intelligent Tutoring Systems (ITSs) (Swartz & Yazdani 92; Holland et al. 95). Despite Natural Language Processing (NLP) technology has significant progress, recent ICALL systems hardly employ NLP techniques to solve the problems of language tutoring. As a result most of the projects in foreign language learning are research prototypes where issues of learner modelling, teaching methodology, and the interface have tended to play a secondary role.

On the other hand the number of NLP researchers who look at Computer Assisted Language Learning (CALL) as a potential application is still very restricted. There are many NLP programs solving various specific CALL-relevant problems: lexical thesauri, parsers, corpus, generators, semantic interpreters, speech syntesizers (Tufis 96). However, a few works show how NLP technologies could be used in the language learning (Wilks & arwell 92; Miller & Fellbaum 92; Abeille 92) and only the first of them presents real NLP applications in CALL. As (Zock 96; Kempen 96; Hamburger 96; Tufis 96) point there is a communication gap between NLP and CALL which has to be filled.

In this paper an application oriented approach is suggested that concerns adopting of some NLP techniques in a vocabulary CALL system aimed at teaching Bulgarians in English Computer Scence terminology. The discussion relates to lexicon, word formation rules and term semantics.

Lexicon is a core part of the expert knowledge in ITSs for vocabulary learning. Corresponding to its traditional function as a tool of references the lexicon here is a source for error diagnosing and generating exercises and feedback (see Section 3).

Understanding the meaning of affixes and the way they are used to build words is extremely useful in tackling new lexical items. Section 4 discusses application of morphological rules for generating word formation exercises and feedback.

Similarly to the knowledge-based mashine aided translation systems, e.g. DB-MAT (Angelova & Bontcheva 96), we present explicitly term semantics. Our model is based on conceptual graphs (CGs) (Sowa 84). In section 5 we discuss advantages and applications of this model as a part of the expert knowledge in CALL.

In order to diagnose semantic errors the conceptual distance between terms has to be determined. Similar question arises in some NLP systems, e.g. (Agirre & Rugai 95) describe an approach for word disambiguation based on conceptual distance sensitive to the length of the shortest path that connects the concepts and the depth in the hierarchy. Section 5 presents our approach based on CGs model for measuring semantic distance and diagnosing in CALL.

2 Overview of ITELS[1]

The work in the project ITELS was motivated by the needs of the English instruction for the technical university students in Bulgaria. The experience shows that the students need support both in the foreign language learning and in learning the terminology, i.e. subject area concepts. In addition, translators who lack knowledge in the subject area while translating technical texts also need help in

[1]ITELS is an acronym for Intelligent Terminology Learning System. The system is described in more details in (Dicheva & Dimitrova 96, Dimitrova & Dicheva 97).

the understanding of the domain concepts.

ITELS is a knowledge-based system aimed at assisting foreign language learning focused on specific terminology as well as enhancing the knowledge of the concepts in the subject area. Basic learner's skills pursued in ITELS are reading and comprehension of English terminological texts as well as understanding and correct usage of subject area terminology. Consequently, the system adopts reading comprehension (Nuttal 82) and vocabulary learning (Carter&McCarthy 88). The emphasis is on the vocabulary acquisition as mastering the terminological vocabulary has a crucial role in technical texts understanding.

As a contrast to some vocabulary CALL systems (e.g. Ingraham et al. 96; Swartz 92) which are built as learning environments, ITELS supports three styles of tutoring:

- *System-Initiated*: The system decides which teaching activity is most appropriate in the current situation.
- *Collaborative*: A mixed-initiative approach is adopted when the system and the learner work in collaboration.
- *Learner-Initiated*: The system behaves as a learning environment suplying different learning activities.

ITELS's tutoring environment provides the learner with the following components:

- *Reading text*

At the beginning of each learning session the learner is suggested a reading text prepared by the teacher who determines also key terms in each text.

- *Exercises*

The learner is suggested learning blocks (LBs): training and information blocks. Each LB is to be built around a subject area term. The training blocks contain questions to be presented to the learner. Four basic types of *questions* are included: multiple choice, multiple answer, fill-in-the-gap, and matching phrases. Each training block is aimed at mastering one of the following *sub-skills*: understanding the main idea of the text being read, understanding a particular paragraph, mastering the word forms of a lexeme and their correct usage, acquisition of term phrasal structure, term definitions, and relationships between concepts. LBs could be prepared by the teacher or dynamically generated by the system. The order of presenting learning blocks as well as the term to be exercised is determined by the system. For the texts entered by the learner, editing and spell-checker facilities are provided.

- *Feedback*

Except the information blocks preliminary defined by the teacher the learner can receive system generated feedback including hints and explanations.

ITELS's domain knowledge presents the expert's intelligence about the language (lexicon and word formation rules) as well as about the term semantics (conceptual graphs knowledge base). This knowledge provides a source for diagnosing the learner's knowledge and guiding the instruction.

3 Lexicon

ITELS's lexicon contains entries for ordinary words and terms. The ordinary words have part of speech, translations in Bulgarian, and information about inflections. The entry for each term includes its definition, phrasal structure (pattern), translation, and a link to the knowledge base. In order to avoid term ambiguity for terms with several meanings we encode one entry per meaning. Figure 1 summarises the main structure of lexical entries for terms. Other information about terms (e.g. inflections) is kept in ordinary entries.

Entry_ID:	T#m_1
Entry:	'output'
Pattern:	n
Sense_Label:	'data'
Translation:	'izhodni danni'
Definition:	'The result of data processing activity hen it is presented external to the system.'
Concept_ID:	C#k_1

Entry_ID:	T#m_2
Entry:	'output'
Pattern:	n
Sense_Label:	'signal'
Translation:	'izhoden signal'
Definition:	'A signal that is obtained from an electrical circuit, such as a logic circuit'
Concept_ID:	C#k_2

Entry_ID:	T#m_3
Entry:	'output'
Pattern:	n
Sense_Label:	'process'
Translation:	'izwegdane na danni'
Definition:	'To produce a result.'
Concept_ID:	C#k_3

Figure 1: A simplified example of lexical entries[2].

As a part of the Expert model the lexicon provides knowledge for diagnosing and tutoring.

[2] The term '*output*' has three meanings '*data*', '*signal*', and '*process*' (Dictionary of Computing 90).

3.1 Providing diagnostic knowledge

In order to diagnose learner's knowledge the system compares his answers against the correct ones. Traditional language errors are spelling mistakes. The role of the spell checker as a part of the diagnostic procedure in ITELS is only to recognise incorrect term spelling and to suggest an appropriate feedback. ITELS's spell checker detects typographical errors such as letter reversals, insertions, and omissions by applying simple pattern matching techniques.

Usually terms are formed by combining several words. It is frequently the case that compound patterns which are well-formed in Bulgarian are different from those in English. Consequently, the learners often confuse the term patterns. For instance, in our experiments some students translated '*object program*' (n/n) as '*program of objects*' (n/prep/npl) by analogy with '*data structure*' (n/n) that is translated in Bulgarian as '*structure of data*' (n/prep/npl)). The lexicon is used as a source for detecting such errors by comparing the terms patterns.

Translators without enough knowledge in the subject area often confuse special meaning of term components with their traditional meaning in the language (e.g. in our experiments such terms were '*odd*', '*routine*', etc.). On the other hand the learners who are not experts in language often cannot understand a term because they do not know the traditional meaning of its components (e.g. in our experiments the students were confused with '*data capture*' because they didn't know the meaning of '*capture*'). The above confusions might occur when any term components are also presented in the lexicon as ordinary words. ITELS offers the learner such components and requests their meaning (the translation in Bulgarian).

3.2 Providing instructional knowledge

- *Suggesting feedback*

It is often the case that the learner knows the meaning of the expected answer, i.e. he/she knows it in the native language but does not know its translation in the target language. Therefore the learner is allowed to supply the anticipated term in Bulgarian and to ask about its translation in English. So, instead of presenting series of useless and boring questions to teach him/her the term meaning, the system could just suggest the appropriate translation in English. This information is extracted from the lexicon. Again from the lexicon is extracted the information about the phrasal structure of terms and their definitions.

In order to understand fully the content of the drills the learner needs support with translations of the unknown words which are usually ordinary words. These translations are also extracted from the lexicon.

- *Generating exercises*

The lexicon is a source for tests about term definitions and term patterns. They are generated by using some templates. For example, Figure 2 shows a question concerning the definition of the term '*output* (*data*)' which is generated by combining the definition with different terms from the lexicon.

term: *output* (*data*)
kind: *multiple choice*
type: *term definition*
content: Chose the term which relates to the following definition:
'*The result of data processing activity when it is presented external to the system*'
- output device,
- computer,
- output (data),
- data structure.

Figure 2: An example of an exercise about term definition.

4 Word Formation

In ITELS the information about the affixes is represented in a *table of affixes* which contain affix rules similar to these described in (Byrd 83). Each affix rule is presented by *affix*, its *kind* (prefix/suffix), *type* (the affix's purpose), *meaning* (the affix's meaning), *condition* (the subcategorisation and selectional constraints on the base), and *result* (the categorial characteristics of the result after applying the affix). Figure 3 shows some examples of affix rules.

(***un***, *prefix*, '*negative/positive*', '*not, not good enough*', {}, {})

(***mini***, *prefix*, '*size*', '*small*', {}, {})

(***ation***, *suffix*, '*noun-forming*', '*in the manner of*, *verb*, *noun*)

(***er***, *suffix*, '*noun-forming*', '*a person who* / *a thing which*', *verb*, *noun*)

(***al***, *suffix*, '*adjective-forming*', '*having the quality of*', *noun*, *adjective*)

Figure 3: Some affix rules.

The information in the lexicon together with the word formation rules are sources for generating exercises and feedback. A morphological engine is aimed at processing word formation knowledge. It works in two modes:

[1] *Input*: a base and an affix.

Output: a word from the lexicon that is derivable

by applying this affix to the base or NIL
Example: *compute , -er → computer*
compute , -or → NIL.
Description: If the base corresponds to the condition for applying the affix, the engine adds the affix to the base and searches for the generated word in the lexicon.

[2] *Input*: a word.
Output: the base and the affixes from which the word is derived or NIL.
Example: *computer → compute, -er*
computational → compute, -ation, -al
compute → NIL.
Description: The engine seeks for the affix by pattern matching and compares the affix's result with the category of the input word. Then removes the affix and finds the base that corresponds to the affix's condition. The algorithm is repeated recursively.

- *Generating exercises*

The system uses the output of the morphological engine for generating word formation exercises concerning mainly terms from the subject area. The following templates are used:

[1] An affix (e.g. *-er*) and a number of bases (e.g. *operate, compute, disk, compile, print*) are supplied. The task is to indicate which of the bases can take the affix.

[2] A base (e.g. *compute*) and several affixes (e.g. *un-, -ation, -er, -ment, -ness*) are supplied. The task is to find out which of the affixes can be taken by this base.

[3] A list of affixes of similar function (e.g. the noun-forming suffixes *-er, -or, -ness, -ance*) and a list of bases are supplied. The task is to match bases with affixes.

- *Generating feedback*

The system generates appropriate feedback to wrong answers of word formation exercises. Error can be due to a mistaken match between a base and an affix. In order to explain that the system uses the information supplied by the morphological engine. Two types of errors could be distinguished and explained: (1) the base does not match to the characteristics of the condition in the affix's record (e.g. *disk, -er*) and/or (2) the base matches to the characteristics of the condition in the affix's record but the lexicon does not contain the word derived from this base and affix (e.g. *operate, -er*).

5 Term Semantics

Terms are linguistic entities which name concepts with a special meaning in the subject area. In ITELS, the term semantics is represented by conceptual graphs (Sowa 84). Conceptual Graphs are formalism with direct mapping to natural language and allow convenient representation of semantics (Sowa 92). As a part of language domain they have been already used in CALL, e.g., in SWIM CGs represent the meanings of the sentences and supply advantageous environment for acquiring the relations between sentence form and meaning (Zock 92).

In ITELS conceptual graphs represent subject area concepts and relations between them[3]. A conceptual graph connects concepts with conceptual relations. An example of a CG that presents the sentence "*An object program is the translation of a source program into an object language*" is shown in Figure 4.

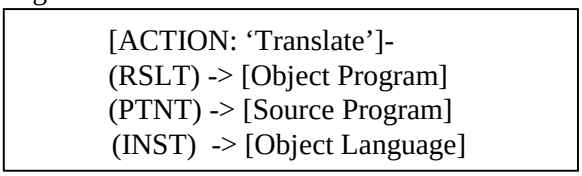

Figure 4: An example of a CG.

Each concept from the subject area is determined by its *concept type* and its *referent*. The concept types are organised in a hierarchy according to their level of generality. Part of the type taxonomy is shown in Figure 5.

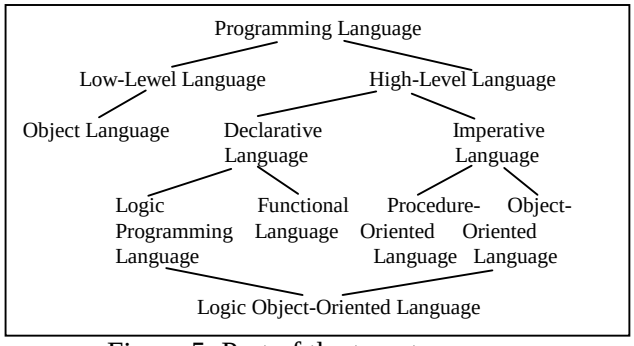

Figure 5: Part of the type taxonomy.

Referents represent specific individuals. Some of the concepts do not identify a particular individual, they are *generic concepts*. *Individual concepts* refer to a particular individual. A set of basic conceptual relations which are widely used in terminological areas are used in ITELS. They correspond to the relations described in (Sowa 84). By applying CGs' operations *restrict*, *join*, and *simplify* (Sowa 84) we obtain new conceptual graphs representing deeper relationships between terms. The *projection mapping* (Sowa 84) allows extraction of information

[3]The conceptual knowledge is based on (Dictionary of Computing 92, Ignatova 92, and Boeckner & Brown 94).

with specific properties from a CG (e.g. all attributes, characteristics, etc.). New types of concepts and relations can be defined in terms of simpler ones. A new type is defined by specifying its supertype (*genus*) and a defining graph (*differentia*) that allows the new type to be distinguished by the genius. New conceptual relations can also be defined by a CG from more basic relations.

In ITELS CGs are source for detecting and correcting learner misconceptions, and also for selecting the term to be exercised. CG operations enable knowledge inference and thus support the generation of exercises and feedback. CGs can underlie various exercises which is extremely useful when a learner faces regular difficulties with some terms.

5.1 Providing knowledge for answer evaluation

When comparing the learner's answer against the correct one, the system determines *conceptual distance* between them. The conceptual distance depends on the place of the concept types in the taxonomy and the participation of the concepts in the CGs.

Two terms are considered to be *hierarchically near* if they are connected by a direct link or the length of the path from each concept type to their common supertype is less than two. For example, as shown in Figure 5 the term '*object-oriented language*' is hierarchically near to all terms with types placed in a sub-tree with root '*high-level language*'.

Two terms are *conceptually near* if they are in the same CG or such a graph could be obtained from the CG knowledge base by using some operations. For example, the terms '*object program*' and '*translation*' are *conceptually near* (see Figure 4).

Two terms are *far* if they are not near. For example, terms '*object-oriented language*' and '*object language*' are *far*.

5.2 Providing instructional knowledge

- *Generating Feedback*

In order to suggest hints the system finds *similar* terms, i.e. *conceptually* and *hierarchically near* to the anticipated answer.

The system explains the difference between a concept type and its *genus* by using the *differentia* from the type definition. For example, corresponding to the definition shown in Figure 6, the difference between '*declarative language*' and '*functional language*' is explained by generating the sentence '*The functional language is a declarative language which operates with functions*.'

type Functional Language (*x*)
genus: Declarative Language
differntia:
[Declarative Language *x]-
(INST) -> [Function {*}]

Figure 6: A definition of the new concept type *Functional Language*.

When two terms are *conceptually similar* the system uses a *projection mapping* for generating the explanations. Some specific relationships between concepts could be included. For example, about '*object-oriented language'* and '*inheritance*' that are *conceptually similar* as shown on Figure 7 the system could generate the following feedback:

'The terms 'object-oriented language' and 'inheritance' are semantically close. 'Inheritance' is an attribute of an object-oriented language. The other attributes are 'polymorphism', 'data abstraction', and 'encapsulation'.

[Object-Oriented Language]-
(ATTR) -> [Polymorphism]
(ATTR) -> [Inheritance]
(ATTR) -> [Data Abstraction]
(ATTR) -> [Encapsulation]
(INST) -> [Object : {*}]

Figure 7. A CG that is a source for the explanation about the difference between '*object-oriented language'* and '*inheritance*'.

- *Generating Exercises*

For the exercises about term understanding the system extracts conceptual knowledge from the CGs KB and generates sentences either directly from a CG or from a CG obtained by *joint*, *restrict*, and *simplify*. For example Figure 8 shows an exercise generated from the CG shown in Figure 4.

term: *object program*
kind: *fill_in_the_gaps*
type: *term understanding*
content: Fill in the blank the correct term:
"An ..... is the translation of a source program into an object language."

Figure 8: An exercise generated from the CG shown in Figure 4.

- *Selecting Terms to be Exercised*

For each selected course topic, the system maintains a list of terms to be exercised. This list contains initially all key terms of the topic. The system updates this list after each answer. In case of erroneous answer it includes new terms in the list

using information from two sources - the KB and the currently used LB - depending on the error type. Assume, for example, that in a LB the learner has suggested the erroneous term "*object program*" which is *conceptually near* to the correct term "*source program*". The system will then include all terms from the CG containing both terms, namely "*object language*" and "*translation*". If both terms - the correct one and the erroneous one are *near* in the taxonomy and have common parents these parent terms will be also included in the list. If both terms are found to be *far* the terms from the corresponding LB would be included in the terms list.

6 Conclusions

ITELS is under development. A simple prototype is completed and is planed to be used in English courses for students of informatics at the University of Shumen. The future work includes completing the system as well as improving some modules. For a more deep error detection concerning cognitive mechanisms about learner's misconceptions, it is worth to explore the relationships between a spelling error and the term components as well as between sound and text. At the present state of development the engine that generates sentences from CGs uses a simplification of the algorithm described in (Sowa 84). Other approaches to generation (e.g. Angelova & Bontcheva 96; Nicolov et al. 95) will be also considered.

Acknowledgements

We are extremely grateful to Galia Angelova and Kalina Bontcheva for their stimulating discussions and help in preparing this paper. We wish to thank also all people from Shumen University who helped in the project developing. The authors wish to thank the anonymous referees who provided helpful comments on the paper.

This work is partly funded by NSF Project No. MY-MM 6/95 and NSF Project No. I-502/95.

References

(Abeille 92) A. Abeille. A Lexicalaized Tree Adjoining Grammar for French and its Relevance to Language Teaching. In M. Swartz and M. Yazdani, editors, *Intelligent Tutoring Systems for Foreign Language Learning*, Springer-Verlag, pages 65-88, 1992.

(Agirre & Rigau 95) E. Agirre and G. Rigau. A Proposal for Word Sense Disambiguation using Conceptual Distance. In *Proceedings of RANLP'95*, pages 258-264, Tzigov Chark, Bulgaria, September 1995.

(Angelova & Bontcheva 96) G.Angelova and K. Bontcheva. NL Domain Explanations in Knowledge Based MAT. In *Proceedings of COLING'96* , pages 1016-1019, Copenhagen, Denmark, August 1996.

(Boeckner & Brown 94) K. Boeckner and P. Brown. *Oxford English for Computing*, Oxford University Press, Oxford, 1994.

(Byrd 83) R. Byrd. Word Formation in Natural Language Processing Systems. In *Proccedings of the Eight International Joint Conference on Artificial Intelligence*, William Kaufmann Inc., Los Altos, pages 704-706, 1,1983.

(Carter & McCarthy 88) R. Carter and M. McCarthy. *Vocabylary and Language Teaching*, Longman, London, 1988.

(Dicheva & Dimitrova 96) D. Dicheva, V. Dimitrova. Representation and Extraction of Terminological Knowledge in ITELS - an Intelligent System for Foreign Language Terminology Learning Terminology Learning. In *Proceedings of JCKBSE'96*, pages 90-91, Sozopol, Bulgaria, September 1996.

(Dictionary of Computing 90)*Dictionary of Computing*, Oxford University Press, Market House Books Ltd., 1990.

(Dimitrova & Dicheva 97) V. Dimitrova, D. Dicheva. Who is Who: The Roles in an Intelligent System for Foreign Language Terminology Learning. In *Proceedings of PEG'97*, Sozopol, Bulgaria, June, 1997 (to appear).

(Hamburger 96) H. Hamburger. Yes! NLP-based FL-ITS will be Important. In *Proceedings of COLING'96* , pages 1007-1009, Copenhagen, Denmark, August 1996.

(Holland et al. 95) M. Holland, J. Kaplan, and M.Sams, editors. *Intelligent language Tutors: Theory Shaping Technology*, Lawrence Erlbaum, 1995.

(Ignatova 92) E. Ignatova. *Computer English*, Tehnika, Sofia, 1992 (in Bulgarian).

(Ingraham et al. 96) B. Ingraham T. Chanier, and C. Emery. CAMILLIE an European project to develop language training for different purposes in various languages on a common hypermedia framework. *Computers and Education*, pages 107-115, 23, (1-2), 1994,.

(Kempen 96) G. Kempen. Human Language Technology can Modernize Writing and Grammar Instruction. In *Proceedings of COLING'96* , pages 1005-1006, Copenhagen, Denmark, August 1996.

(Miller & Fellbaum 92) G. Miller and C. Fellbaum. WordNet and the organisation of Lexical Memory, In M. Swartz and M. Yazdani, editors, *Intelligent Tutoring Systems for Foreign Language Learning*, Springer-Verlag, pages 89-102, 1992.

(Nicolov et al. 95) N. Nicolov, C. Mellish, and G. Ritchie. Sentence Generation from Conceptual Graphs. In G. Ellis, R. Levinson, W. Rich, and J. Sowa, editors. *Conceptual Structures: Applications, Implementation and Theory Proceedings of ICCS'95*, pages 74-88, Berlin, Springer, 1995.

(Nuttal 82) C. Nuttal. *Teaching Reading Skills in a Foreign Language*, Heinemann, Oxford, 1982.

(Sowa 92) J. Sowa. Conceptual Graphs Summary, in T. Nagle, G. Nagle, L. Gerholz, and P. Eklund, editors, *Conceptual Structures: Current Researches and Practice*, Ellis Horwood, pages. 3-52, 1992.

(Sowa 84) J. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, MA, 1984.

(Swartz & Yazdani 92) M. Swartz and M. Yazdani, editors. *Intelligent Tutoring Systems for Foreign Language Learning*, Springer-Verlag, 1992.

(Swartz 92) M.Swartz. Issues for Tutoring Knowledge in Foreign Language Intelligent Tutoring Systems. In M. Swartz and M. Yazdani, editors. *Intelligent Tutoring Systems for Foreign Language Learning*, Springer-Verlag, pages 219-234, 1992.

(Tufis 96) D. Tufis. CALL: The Potential of Lingware and the Use of Empirical Linguistic Data. In *Proceedings of COLING'96* , pages 1010-1011, Copenhagen, Denmark, August 1996.

(Wilks & Farwell 92) Y. Wilks and D. Farwell. Building an Intelligent Second Language Tutoring System from Whatever Bits You Happen to Have Lying Around. In M. Swartz and M. Yazdani, editors, *Intelligent Tutoring Systems for Foreign Language Learning*, Springer-Verlag, pages 263-274, 1992.

(Zock 92) M. Zock. SWIM or Sink: The Problem of Communicating Thought. In M. Swartz and M. Yazdani, editors, *Intelligent Tutoring Systems for Foreign Language Learning*, Springer-Verlag, pages 235-248, 1992.

(Zock 96) M. Zock. Computational Linguistics and its Use in Real World: The Case of Computer Assisted Language Learning. In *Proceedings of COLING'96* , pages 1002-1004, Copenhagen, Denmark, August 1996.