NLP-Supported Full Text Retrieval

Michael PIOTROWSKI

Abstract. This article analyzes the usefulness of NLP techniques for text retrieval and presents a new research project, which tries to take into account the results of the analysis, namely by replacing stemming with full morphological analysis.

Résumé. Cet article analyse les méthodes de traitement automatique du langage naturel (TALN) en ce qui concerne leur utilité pour la recherche textuelle. Il décrit un projet de recherche nouveau qui essaie de prendre en considération les résultats de cette analyse, particulièrement en remplaçant la « troncation » par une véritable analyse morphologique.

Keywords: Text retrieval, natural language processing (NLP), morphology.

Mots-clés : Recherche textuelle, traitement automatique du langage naturel (TALN), morphologie.

1. What is Information Retrieval?

While information can be encoded in many forms, most of the world's information is probably stored in documents written in natural language. Two major tasks for information retrieval (IR) are therefore *text retrieval* (as opposed to the retrieval of other forms of information) and *document retrieval* (*i.e.* the retrieval of full documents, as opposed to, e.g., automatically generated abstracts). Note that text retrieval and document retrieval are not synonyms but sub-areas of IR, although this distinction is not always observed.

✉ Abteilung für Computerlinguistik; Friedrich-Alexander-Universität Erlangen-Nürnberg; Bismarckstraße 12; D–91054 Erlangen (Germany).
Fax: +49 91 31 85 92 51 E-mail: mxp@linguistik.uni-erlangen.de

1.1. The Traditional Approach

A traditional text retrieval system works as follows: first, all documents to be retrieved—the so-called *document collection*—are scanned in various ways for *indexing terms*, usually words or phrases. The indexing terms are supposed to represent the content of the document. A user query is processed in a similar way to extract *query terms* which are then matched against the indexing terms. In response to the query the system returns a set of references to documents in the document collection which are considered relevant to the query.

A system working exactly as described above would not yield optimal results. Quite early on in the development of text retrieval systems, methods were devised to improve the search results, and especially to cope with the problem of *vocabulary mismatch* (described below). Apart from statistical enhancements, like relevance feedback, which are not the topic of this article, common extensions are *stop-word lists*, *thesauri* and *stemming*.

1.2. Retrieval Enhancements

While natural language is the encoding most easily understood by humans, it can pose serious problems for machines. Natural language is characterized by a high degree of redundancy on the one hand, but on the other hand, much of the information is only implicit and can only be understood with a knowledge of the world. For example, users of IR systems are not looking for strings of characters but for concepts. If a user issues a query for *cars*, it normally doesn't mean that they are interested in every occurrence of the string "c-a-r-s", but rather in documents concerned with the concept of self-propelled land vehicles, which can be partially—not exhaustively—described with the words *car*, *automobile*, *pickup*, *minivan*, but maybe also by *BMW*, *Hyundai*, or *set of wheels*. A text about cars obviously doesn't need to mention the word *cars* even once. On the other hand, despite the high frequency of *cars* in this text, it obviously isn't a text about cars.

The problem that the user uses a word or wordform in their query that doesn't occur in the text, is referred to as *vocabulary mismatch*. Stemming (the "cutting off" of affixes) tries to account for morphological variation while thesauri are used to handle lexical variation.

Some people would describe the use of stemming and thesauri as natural language processing (NLP) but in fact stemming is not really

linguistically motivated, and it covers only a tiny fraction of morphology. A lot of work has already been done on the effects of stemming on IR performance, unfortunately most of it for English, which has a relatively simple morphology. Consequently, it is still disputed whether stemming is useful, because the results for English are equivocal. Church (1995) even concludes from these results that all NLP for retrieval is more or less useless:

> These results [claiming that stemming produces little if any improvement in precision/recall] are disturbing for those of us working in natural language processing (NLP). If it is hard to show that something as simple as stemming is helpful, how can we possibly justify our interests in more challenging forms of natural language processing such as part of speech tagging, word sense disambiguation, synonymy, phrase identification and parsing?

There is, however, considerable evidence (Pohlmann: 1997, Sheridan and Ballerini: 1996, Choueka: 1992, Popovic and Willett: 1992) for the usefulness of stemming for other, more inflectional languages. What was also found out in these experiments is that for languages which have a productive composition process, compounds must be handled as well. This shouldn't be a surprise, though, as stemming was designed for English, where affixation is nearly the only morphological process. It is probably desirable to handle affixation and composition (and other morphological processes) in a uniform way on a linguistic basis.

The use of thesauri massively reduces the precision of the IR system and generates hypotheses about the concepts the user is looking for, which may not be true. Furthermore, not only does this reduce precision but it potentially adds highly ranked false matches to the retrieval results where the user can't see why they were retrieved and cannot adapt and refine their queries.

2. NLP in Text Retrieval

2.1. Syntactic and Semantic Analysis

NLP researchers tend to consider conventional retrieval systems "inadequate for the obvious reason that they do not do NLP, and so cannot tell a Venetian blind from a blind Venetian" (Sparck Jones: 1997). The logical consequence is that syntactic and semantic relationships are tried to be made explicit. Actual systems implement this approach to various degrees: The *CLARIT* system and experiments based on it (e.g. Evans and Zhai: 1996) aim just for a shallow understanding of the texts by approximating concepts through the analysis of the phrasal structures of the documents. At the other end of the spectrum lies the *FERRET* project (Mauldin: 1991), which aimed at a higher level of conceptual understanding.

Semantics, however, is a field which is not yet really understood, especially for large quantities of free text. Systems that can handle at least some aspects of semantics are too slow and unstable for industrial-strength IR.

Syntax is much better understood than semantics, and *CLARIT*, for example, is a commercial product, but most of the points which make the use of semantics in production systems unwieldy (too difficult, too large, too slow) also apply to syntactic analysis. Syntactic and semantic analyses are also very expensive to adapt to multiple languages.

Furthermore, the use of syntax and semantics is based on the assumption that an indexing for retrieval must directly and explicitly capture the syntactic and semantic relationships contained in the documents, as it is done in manual indexing. However, it is not certain that this is absolutely necessary; to quote from Sparck again:

> [...] decades of past experiment have shown that complex index descriptions modelled on manual prototypes are far too constraining, while complex terms (e.g. simple phrases with a head-modifier structure) do not work much better than coordinated simple terms. These findings have been confirmed by tests under the current ARPA/NIST Text REtrival Conference (TREC) evaluation programme, where many alternative specific approaches are being assessed using very large full-text files.
>
> (Sparck Jones: 1997, p. 13.)

An additonal point to consider is that both the users and the data of text retrieval systems have changed: the World-Wide Web has made text retrieval an end-user application used to retrieve short-lived

multimedia documents in many languages out of a document collection of nearly infinite size. And if you watch Web queries with WebCrawler's ⟨http://www.webcrawler.com/⟩ *Search Ticker*, it is obvious that one and two word queries are the most common type of query, which doesn't leave much room for syntactic and semantic interpretation.

Although there are advances in the use of syntactic and semantic analysis for use in IR, Salton and McGill's opinion from 1983 still seems to be valid:

> Various attempts have been made to use simple syntactic analysis systems in actual information retrieval situations. While linguistic methods may eventually prove essential in automatic indexing, the evidence available indicates that the simplified syntactic analysis systems do not yet provide the answer. The frequency-based phrase-generation methods are simpler to implement and are currently more effective. (Salton and McGill: 1983, p. 91.)

2.2. Morphological Analysis

Simple syntactic analysis may be useful for languages such as English, where composition is mostly a syntactic process. In languages such as German, however, where compounds are written as one orthographic unit, morphology seems to be of much more immediate value.

An approach that—to my knowledge—has not yet been evaluated, is not to use syntactic analysis, but to replace stemming—which is already known to be useful—with full morphological analysis. Morphology has long been well understood in traditional linguistics as well as in computational linguistics, and there are implementations of morphological grammars which are potentially fast and stable enough to be used on a day-to-day basis in a real-world IR system. Using morphology for text retrieval has—at least theoretically—all the advantages of stemming, and it is based on linguistic principles. If one is planning to use syntactic and semantic analysis one day, morphological analysis will be an essential precondition anyway.

3. Multilingual Aspects

3.1. Types of Multilingual Support

In an increasingly global marketplace, there is another important point to consider: multilinguality. Regarding their support for multiple languages, IR systems can be classified into 3 categories, as illustrated in figures 1, 2 and 3:
— language-dependent,
— language-independent,
— multilingual.

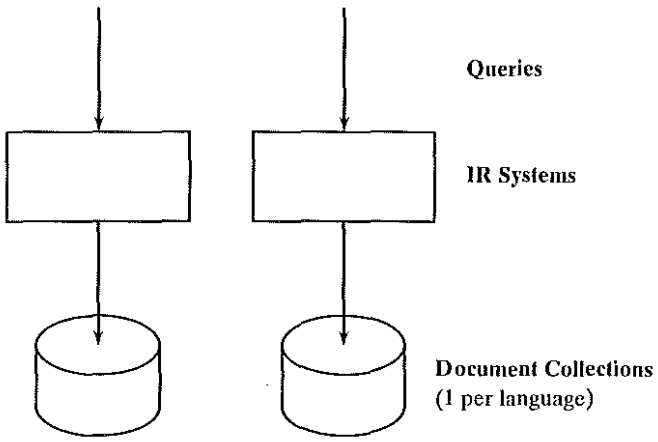

Fig. 1.– Language-dependent IR

NLP-supported text retrieval is inherently language-dependent; after all, NLP is always language-dependent. Language-dependent retrieval systems are of limited practical use, though, because most document collections today contain documents in more than just one language.

Language-dependent IR systems are theoretically able to achieve very good search results for the language they're designed for. However, if you have documents in more than one language, you're forced to use different systems to retrieve these documents, and you have to store the documents in different collections. Otherwise, the retrieval system used would have to identify the language of the documents to see if they are equipped to handle it. If this is not done, *i.e.* if you run language-dependent IR systems on mixed-language document collections, and the

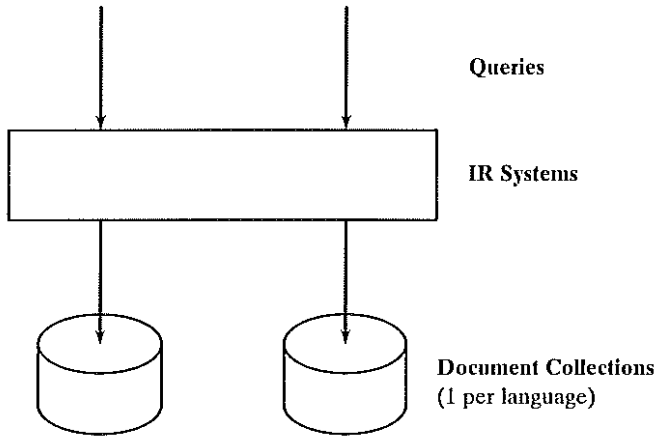

Fig. 2.– Language-independent IR

system is not aware that some documents are not written in the language it knows, the results are unpredictable.

Conventional retrieval systems might be considered mostly language-independent, although most systems now contain language-dependent enhancements like stop-word lists or stemmers. A fully language-independent system can do nothing more refined than simple pattern matching, thus delivering only average performance for all languages (assuming that this works equally well for all languages, which is not the case). And of course, you have to formulate different queries to retrieve documents in different languages; however, you do not have to use different retrieval systems, as is the case for language-dependent IR, but you do not get optimal results, either.

Truly multilingual IR is the most demanding form of IR. A multilingual IR system has to support at least two languages, and you can enter your query in any of the supported languages and get all matching documents regardless of their language. For example, with a multilingual IR system for German, English and French you could search for *cars*, and also get German documents containing *Auto* and French documents containing *voiture*.

3.2. NLP and Multilinguality

Syntax and semantics are probably too expensive if they must be provided for every supported language; morphology, however, seems to be useful and not too difficult to be implemented for most languages.

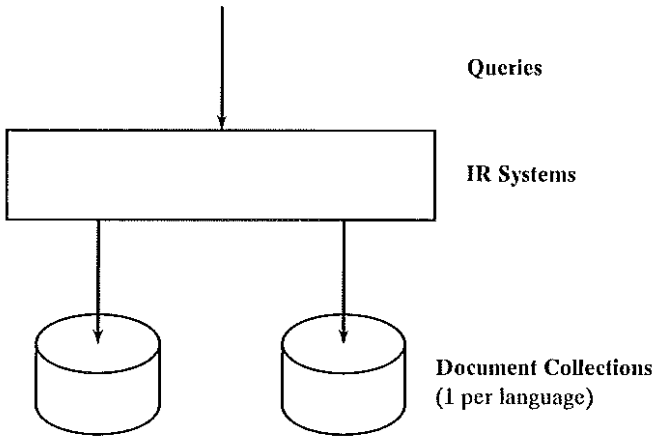

Fig. 3.– Multilingual IR

Language-dependent IR systems are not commercially interesting anymore, except perhaps for some very specialized applications. Language-dependent IR systems force their users to use different systems for other languages, which is unacceptable. Most users would probably prefer a language-independent IR system, which yields worse results, but they wouldn't have to learn a different system.

The most desired—but not yet widely available—form of IR, however, is certainly multilingual IR. NLP might actually help here: you need some kind of language understanding anyway, so using true NLP is probably the best solution.

4. The IRF/1 Approach To NLP-Supported IR

IRF/1 is a new research project at the Department for Computational Linguistics at the University of Erlangen (CLUE), and will be mainly developed in the author's master's thesis. The main goal of the IRF/1 project is to evaluate whether the use of NLP techniques, especially morphology, can improve the retrieval performance and/or user-friendliness of a text retrieval system. Another important aspect is to test the suitability of the NLP components for use in the text retrieval environment, especially with regard to their ability to:

1) Process large amounts of text,
2) Process unrestricted text. (See Evans and Zhai: 1996.)

IRF/1 tries to take into account the problems outlined above: It doesn't try to understand the texts, but rather tries to optimize the "text processing" part of the retrieval process by employing morphological analysis based on "hard" linguistics. Because it is based on linguistic principles which are applicable to almost all languages, it will be possible to implement support for multiple languages in a uniform way.

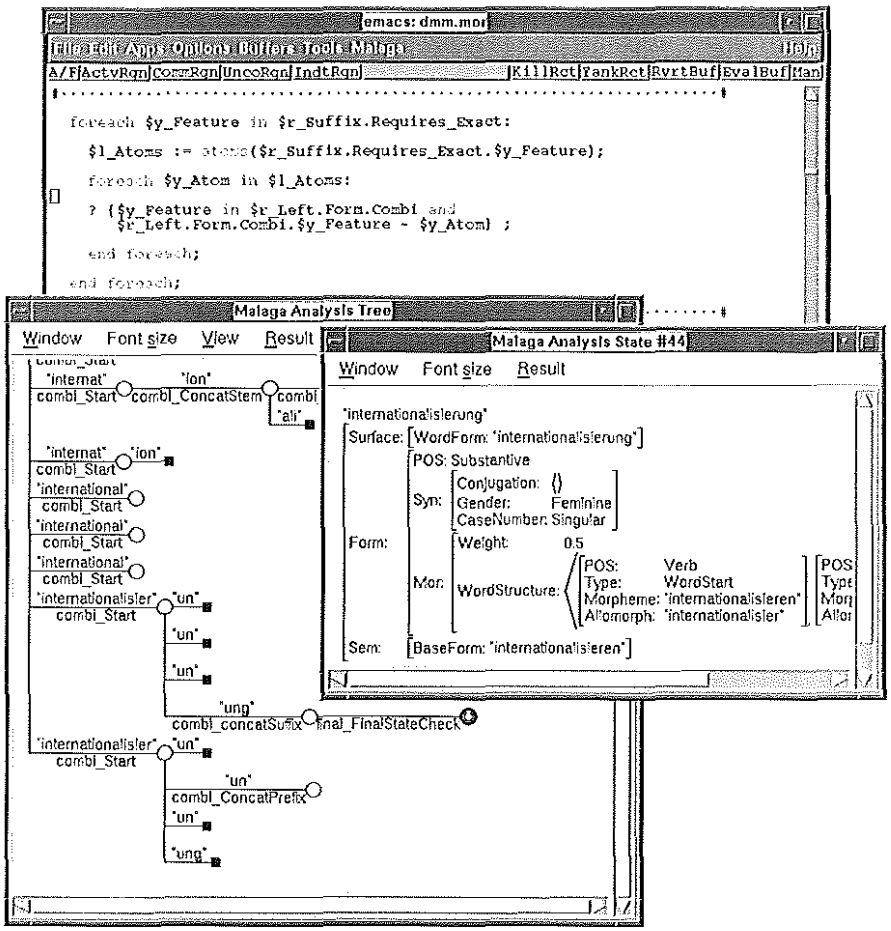

Fig. 4.– Malaga Development Environment

The basis for the linguistic components of IRF/1 is the *Malaga* grammar development system (Beutel: 1997, Schulze: 1997). *Malaga* was developed at CLUE and consists of a specialized programming

language for natural language grammars, a compiler for that language, a development environment including debugging facilities, a visualization tool (see figure 4), and libraries for using *Malaga* grammars from C and Perl applications. *Malaga* is based on the formalism of *Left-Associative Grammar* (Hausser: 1989, Hausser: 1992), which is characterized by formal simplicity and computational efficiency, while being linguistically well-motivated. The available grammars written in *Malaga* currently include morphology grammars for German, Italian and Korean.

Malaga is freely available for research purposes, and you are encouraged to download the current version from our FTP server at `ftp://ftp.linguistik.uni-erlangen.de/` to try out it for yourself.

IRF/1 will, in some ways, be similar to other current European research Projects. Table 1 contains a point-for-point comparison.

Table 1

Comparison of Text Retrieval Research

Area	Systems		
	UPLIFT	*EuroSPIDER*	IRF/1
NLP	stemming, dictionary-based stemming	stemming, dictionary-based stemming	morphology
Phrase indexing	no, rejected	planned	no
Dictionary	CELEX Dutch	CELEX German	Proprietary
Dictionary size (stems)	124,000	51,000	49,000
Dictionary size (wordforms)	380,000	360,000	unlimited
Query processing	expansion	normalization	normalization
Multilingual	planned	yes	planned
Uniform approach for all languages	N/A	no	planned
Automatic language identification	no	no	planned
Fallback for unsupported languages	N/A	N/A	planned
Available	no	yes, commercial	no

Conceptually, the IRF/1 approach can be described as follows (see figure 5): the incoming documents are analyzed morphologically and the base forms of the words in the documents are stored in the index. The procedure for a query is similar: the words of the query are reduced to their base forms and these are then looked up in the index.

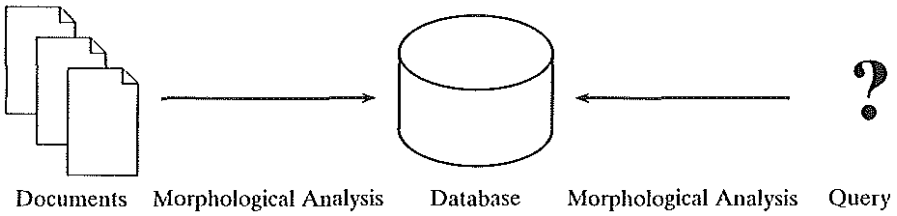

Fig. 5.– Conceptual View of IRF/1

Although morphological analysis is used, there will be no syntactic analysis. This has several reasons:

— according to Salton (Salton and McGill: 1983), and more recently, Sparck Jones (Sparck Jones: 1997) syntactic analysis does not significantly improve search results;
— in the *UPLIFT* project (Pohlmann: 1997) it was found that it is possible to somewhat improve precision by using syntactic analysis, but that it adds too much computational overhead to be useful for WWW applications;
— typical end-user queries are too short for syntactic analysis.

At the moment it seems to be more sensible to try to use morphological analysis in the places where stemming is currently the most popular method. Stemming is not really NLP, and especially the handling of compounds is often stacked on top of it instead of being integrated. The *UPLIFT* and *EuroSPIDER* projects proved that the recognition of compound constituents is important for both Dutch and German. If you consider compounds like *Lebensversicherungsgesellschaft* (German for *life insurance company*), it becomes clear why it is necessary for retrieval systems to be able to identify the constituents. IRF/1 will address this by applying "deep" morphological analysis to the documents.

As you can see from table 1, a normalization approach will be taken for query processing. This means that only base forms will be stored in the index, which can greatly decrease its size; this is important when document collections become very large. Since morphological analysis

provides information about the part of speech (POS) or word class of a word, it is possible to employ a list of *stop classes*, *i.e.* those word classes which have only or mostly grammatical functions, or are generally useless for retrieval purposes due to their frequency, like determiners or prepositions. Stop classes are similar to stop words—which could be used at the same time—but allow to control the inclusion or exclusion of large numbers of words with a single list entry. Naturally this further reduces the size of the index.

Recognizing that multilingual applications will be rather the rule than the exception in the future, IRF/1 will be designed from the ground up to accommodate multilingual applications. To ensure language independence, IRF/1 will, when completely implemented, automatically detect the language of a document, and accordingly use the language specific methods available for that language, or fall back to a generic mode when none are available. Furthermore, IRF/1 will use the Unicode character set (The Unicode Consortium: 1996) for its index and will be able to convert to and from other character sets for input and output.

5. Conclusions

It is still disputed whether NLP techniques are useful for IR. Different experts in this area often hold widely varying views.

Instead of making yet another attempt at finally determining the usefulness of NLP for text retrieval, the IRF/1 project simply tries to optimize the positive results of stemming for languages morphologically richer than English by replacing it with full morphological analysis.

Although syntactic analysis will not be used in the first version of IRF/1, it will nevertheless be extensible by syntactic grammars written in *Malaga*. By using a grammatical formalism powerful enough to potentially describe all natural languages, and by not restricting the document collection to only contain 8 bit characters, IRF/1 will hopefully be able to lay a solid foundation for future multilingual retrieval applications.

Bibliography

BEUTEL (Björn): 1997, *Malaga 3.0* (Friedrich-Alexander-Universität Erlangen-Nürenberg, user's manual included in the Malaga distribution).

CHOUEKA (Yaacov): 1992, "Responsa: An Operational Full-Text Retrieval System with Linguistic Components for Large Corpora", *Computational Lexicology and Lexicography: a Volume in Honor of B. Quemada*, A. Zampoli (ed.) (Pisa: Giardini Press).

CHURCH (Kenneth Ward): 1995 "One Term or Two?", *Proceedings of the 18th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval 1995*, p. 310–318.

EVANS (David A.) and ZHAI (Chengxiang): 1996, "Noun-Phrase Analysis in Unrestricted Text for Information Retrieval", *34th Annual Meeting of the Association for Computational Linguistics—Proceedings of the Conference* (published by ACL), p. 17–24.

HAUSSER (Roland): 1989, *Computation of Language. An Essay on Syntax, Semantics and Pragmatics in Natural Man Machine Communication* (Berlin–...: Springer Verlag, "Symbolic Computation – Artificial Intelligence").

HAUSSER (Roland): 1992, "Complexity in Left-Associative Grammar", *Theoretical Computer Science* 106 (2), p. 283–308.

LEE (Kiyong): 1997, *Han.kuk.e.ui cwa.yen.cep. mwun.pep kwu.chwuk: myeng.sa e.hyeng.kwa ton.sa.e.hyeng.ui pwun.sek [Construction of a Left-Associative Grammar for Korean. The Analysis of Noun and Verb Forms]* (Korea University, Seoul, Technical Report).

LORENZ (Oliver): 1996, *Automatische Wortformerkennung für das Deutsche im Rahmen von Malaga* (Friedrich-Alexander-Universität Erlangen-Nürnberg, Master's Thesis).

MAULDIN (Michael L.): 1991, *Conceptual Information Retrieval. A Case Study in Adaptive Partial Parsing* (Boston–Dordrecht–London: Kluwer).

POHLMANN (René): 1997, "The UPLIFT Text Retrieval Project", *elsnews* 6.3, p. 9–10.

POPOVIČ (M.) and WILLETT (P.): 1990, "Processing of documents and queries in a Slovene language free text retrieval system", *Literary and Linguistic Computing* 5 (2), p. 183–190.

SALTON (Gerard) and MCGILL (Michael): 1983, *Introduction to Modern Information Retrieval* (New York–...: McGraw-Hill).

SCHULZE (Markus): 1997, "Morphologie, Syntax und Semantik im Rahmen der linksassoziativen Grammatik", to appear in *Proceedings der GLDV-Jahrestagung 1997*, Leipzig.

SHERIDAN (Páraic) and BALLERINI (Jean-Paul): 1996, "Experiments in Multilingual Information Retrieval using the SPIDER System", *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval SIGIR 96* (Konstanz: Hartung-Gorre), p. 58–65.

SPARCK JONES (Karen): 1997, "The way forward in information retrieval", *elsnews* 6.3, p. 12–13.

THE UNICODE CONSORTIUM: 1996, *The Unicode Standard, Version 2.0* (Reading, MA–...: Addison-Wesley).

WETZEL (Christian): 1996, *Erstellung einer Morphologie für Italienisch in Malaga.* (Friedrich-Alexander-Universität Erlangen-Nürnberg, Pre-Master's Thesis).