Chapter X
Standardization of Terms Applying Finite-State Transducers (FST)

Carmen Galvez
University of Granada, Spain

ABSTRACT

This chapter presents the different standardization methods of terms at the two basic approaches of nonlinguistic and linguistic techniques, and sets out to justify the application of processes based on finite-state transducers (FST). Standardization of terms is the procedure of matching and grouping together variants of the same term that are semantically equivalent. A term variant is a text occurrence that is conceptually related to an original term and can be used to search for information in a text database. The uniterm and multiterm variants can be considered equivalent units for the purposes of automatic indexing. This chapter describes the computational and linguistic base of the finite-state approach, with emphasis on the influence of the formal language theory in the standardization process of uniterms and multiterms. The lemmatization and the use of syntactic pattern-matching, through equivalence relations represented in FSTs, are emerging methods for the standardization of terms.

INTRODUCTION

The purpose of a information retrieval system (IRS) consists of retrieving, from amongst a collection of documents, those that respond to an informational need, and to reorganize these documents according to a factor of relevance. This process normally involves *statistical methods* in charge of selecting the most appropriate terms for representing documental contents, and an *inverse index file* that accesses the documents containing these terms (Salton & McGill, 1983). The relationship of pertinence between queries and documents is established by the number of

terms they have in common. For this reason the queries and documents are represented as sets of characteristics or indexing terms, which can be derived directly or indirectly from the text using either a thesaurus or a manual or automatic indexing procedure. In many IRS, the documents are indexed by uniterms. However, these may result ambiguous, and therefore unable to discriminate only the pertinent information. One solution to this problem is to work with multiword terms (or *phrases*) often obtained through statistical methods. The traditional IRS approach is based on this type of automatic indexing technique for representing documentary contents (Croft, Turtle, & Lewis, 1991; Frakes, 1992; Salton, 1989).

Matching query terms to documents involves a number of advanced retrieval techniques, and one problem that has not yet been solved is the inadequate representation of the two (Strzalkowski, Lin, Wang, & Pérez-Carballo, 1999). At the root of this problem is the great variability of the lexical, syntactic, and morphological features of a term, variants that cannot be recognized by simple *string-matching algorithms* without some sort of *natural language processing* (NLP) (Hull, 1996). It is generally agreed that NLP techniques could improve IRS yields; yet it is still not clear exactly how we might incorporate the advancements of computational linguistics into retrieval systems. The grouping of morphological variants would increase the average recall, while the identification and grouping of syntactic variants is determinant in increasing the accuracy of retrieval. One study about the problems involved in using linguistic variants in IRS is detailed by Sparck Jones and Tait (1984).

The term standardization is the process of matching and grouping together variants of the same term that are semantically equivalent. A variant is defined as a text occurrence that is conceptually related to an original term and can be used to search for information in text databases (Jacquemin & Tzoukermann, 1999; Sparck Jones & Tait, 1984; Tzoukermann, Klavans, & Jacquemin, 1997). This is done by means of computational procedures known as *standardization or conflation algorithms*, whose primary goal is the normalization of uniterms and multiterms (Galvez, Moya-Anegón, & Solana, 2005). In order to avoid the loss of relevant documents, an IRS recognizes and groups variants by means of so-called conflation algorithms. The process of standardization may involve linguistic techniques such as the segmentation of words and the elimination of affixes, or lexical searches through thesauri. The latter is concerned with the recognition of semantic variants, and remains beyond the scope of the present study.

This chapter focuses on the initial stage of automatic indexing in natural language, that is, on the process of algorithmically examining the indexing terms to generate and control the units that will then be incorporated as potential entries to the search file. The recognition and grouping of lexical and syntactic variants can thus be considered a process of normalization; when a term does not appear in a normalized form, it is replaced with the canonical form. Along these lines, we will review the most relevant techniques for grouping variants, departing from the premise that conflation techniques featuring linguistic devices can be considered normalization techniques, their function being to regulate linguistic variants.

THE PROBLEM OF TERM VARIANTS

During the first stage of automatic indexing in natural language we encounter a tremendous number of variants gathered up by the indexing terms. The variants are considered semantically similar units that can be treated as equivalents in IRS. To arrive at these equivalencies, standardization methods of variants are used, grouping the terms that refer to equivalent concepts. The variants can be used to extract information in the textual databases (Jacquemin & Tzoukermann,

1999). Arampatzis, Tsoris, Koster, and Van der Weide (1998) identify three main types of variations: *(a)* morphological variation linked to the internal structure of words, by virtue of which a term can appear in different forms (e.g., "*connect*," "*connected,*" "*connecting*," and "*connection*" are reduced to "*connect.*" which is considered to be identical for all these morphologically and conceptually related terms); (b) lexico-semantic variation linked to the semantic proximity of the words, so that different terms can represent the same meaning, and multiple meanings can be represented by the same term (e.g., "*anoxaemia*," "*anoxemia,*" and "*breathing problems*" are reduced to "*breathing disorders*"); and (c) syntactic variation linked to the structure of the multiword terms, where altenative syntactic structures are reduced to a canonical syntactic structure (e.g., constructions that are structurally distinct but semantically equivalent, such as "*consideration of these domain properties*" and "*considering certain domain properties*" are conflated to the single structure "*considering domain properties*").

In most cases, the variants are considered semantically similar units that can be treated as equivalents in IRS (Hull, 1996). To arrive at these equivalencies, standardization methods of variants are used, grouping the terms that refer to equivalent concepts. Standardization methods are applied when the terms are morphologically similar. But when the similarity is semantic, lexical search methods are used. To reduce semantic variation, most systems resort to lexical lookup to relate two words that are completely different in form (Paice, 1996). The problems involved in fusing the lexico-semantic variants remain beyond the scope of the present review.

AN APPROACH TO STANDARDIZATION METHODS OF TERMS

Term standardization methods have essentially been developed for English because it is the predominant language in IR experiments. However, with a view to the reduction of uniterm variants, English features a relatively weak morphology and therefore linguistic techniques are not necessarily the most suitable ones. To the contrary, because English relies largely on the combination of terms, the linguisitc techniques would indeed be more effective in merging multiterm variants. The procedures for the reduction of variants of single-word terms can be classified as: (1) *nonlinguistic techniques*, which are stemming methods consisting mainly of suffix stripping, stem-suffix segmentation rules, similarity measures, and clustering techniques; and (2) *linguistic techniques*, which are lemmatization methods consisting of morphological analysis. That is, term standardization based on the regular relations (RR), or equivalence relations, between inflectional forms and canonical forms, represented in *finite-state transducers* (FSTs).

On the other hand, the multiterms that represent concepts are included among what are known as *complex descriptors*. Fagan (1989) suggests two types of relationships: *syntactic* and *semantic*. First, the syntactic relationships depend on the grammatical structure of these same terms and are represented in phrases. The syntactic relationships are of a *syntagmatic* type, allowing the reduction of terms used in document representation, and their contribution in the IRS is to increase average precision. Second, the semantic relationships depend on the inherent meaning of the terms involved and are represented in the classes of a thesaurus. The semantic relationships are of a *paradigmatic* type, allowing us to broaden the terms used in the representation of the documents, and their purpose in the retrieval systems is to increase average recall. Multiword terms are considered to be more specific indicators of document content than are single words, and for this reason many methods have been developed for their identification. Basically there are two approaches: (1) *nonlinguistic techniques*, which are statistical methods based on the computation

of similarity coefficients, association measures, and clustering techniques by means of word and n-gram cooccurrence; and (2) *linguistic techniques*, which are syntactic methods based on syntactic pattern-matching according to local grammars (LG) represented in *finite-state automata* (FSA) (a LG consists of rigorous and explicit specifications of particular structures**)** and pattern standardization through equivalence relations, established between syntactic structure variants and canonical syntactic structures, represented in FSTs.

The application of standardization techniques to single-word terms is a way of considering the different lexical variants as equivalent units for retrieval purposes. One of the most widely used nonlinguistic techniques is that of stemming algorithms, through which the inflectional and derivational variants are reduced to one canonical form. Stemming or suffix stripping uses a list of frequent suffixes to conflate words to their *stem* or base form. Two well known stemming algorithms for English are the Lovins stemmer (1968) and the Porter stemmer (1980). Another means of dealing with language variability through linguistic methods is the fusion of lexical variants into *lemmas*, defined as a set of terms with the same stem and, optionally, belonging to the same syntactic category. The process of lemmatization, or morphological analysis of the variants and their reduction to controlled forms, relies on lexical information stored in electronic dictionaries or lexicons. One such example is the morphological analyzer developed by Karttunen (1983).

In addition to these approaches, it is possible to group multiword terms within a context, assigning specific indicators of relationship geared to connect different identifiers, so that *noun phrases* (NPs) can be built (Salton & McGill, 1983). NPs are made up of two or more consecutive units, and the relationships between or among these units are interpreted and codified as endocentric constructions, or *modifier-head-structures* (Harris, 1951). When we deal with single-word terms, the content identifiers are known as indexing terms, keywords or descriptors, and they are represented by uniterms. Uniterms may on occasion be combined or coordinated in the actual formulation of the search. When multiword terms or NPs are used for indexing purposes, they can include *articles*, *nouns*, *adjectives,* or different indicators of relationship, all parts of a process known as *precoordination* (Salton & McGill, 1983). In indexing multiword terms, most extraction systems employ *part-of-speech* (POS) taggers, which reflect the syntactic role of a word in a sentence, then gather together the words that are components of that NP (Brill, 1992; Church, 1988; Tolle & Chen, 2000; Voutilainen, 1997).

When standardization algorithms are applied to multiword terms, the different variants are grouped according to two general approaches: term cooccurrence and matching syntactic patterns. The systems that use cooccurrence techniques make term associations through different coefficients of similarity. The systems that match syntactic patterns carry out a surface linguistic analysis of certain segments or textual fragments. In addition to the surface analysis and the analysis of fragments from the corpus, many systems effectuate a POS category disambiguation process (Kupiec, 1992). The syntactic variants identified through these methods can be grouped, finally, in canonical syntactic structures (Schwarz, 1990; Sheridan & Smeaton, 1992; Smadja, 1993; Strzalkowski, 1996).

The recognition and standardization of linguistic structures in IRS is an area pertaining to NLP. Within the NLP understanding of the mathematical modelling of language, there are two clearly distinguished conceptions: *symbolic models* and *probabilistic* or *stochastic models*. These models can be traced back to the contribution of Kleene (1956) regarding finite-state mechanisms and to the work by Shannon and Weaver (1949) on the application of the probabilistic processes to finite automatas. Chomsky was the first to con-

sider automatas as mechanisms characterizing the structures of language through grammars, thereby setting the foundations for the *theory of formal languages* (Chomsky, 1957). Finite-state mechanisms are efficient for many aspects of NLP, including morphology (Koskenniemi, 1983) and parsing (Abney, 1991; Roche, 1996).

STANDARDIZATION OF TERMS THROUGH FINITE-STATE TECHNIQUES

Formal language theory focuses on languages that can be described in very precise terms, such as programming languages. Natural languages are not formal, as no well-defined boundary exists between correct sentences or those that are incorrect. Notwithstanding, formal definitions approximating natural language phenomena can be encoded into computer programs and be used for the automated processing of natural language. Likewise, formal descriptions can be utilized by linguists to express theories about specific aspects of natural languages, including morphological analysis. The most important application of the formal language theory to linguistics came from Chomsky (1957). His basic hypothesis was that the different types of formal languages were capable of modeling natural language syntax. This theoretical foundation beneath formal languages and grammars has a direct relation with the theory of machines or *automata*, abstract devices able to receive and transmit information. A finite automata accepts a string or a sentence if it can trace a path from the initial state to the final state by jumping along the stepping stones of *labeled transitions*. A finite automata is thus defined as a network of states and transitions, or edges, in which each transition has a label (ROCHE, 1996). Formally, a FSA is a 5-tuple:

$FSA = <\Sigma, Q, q_0, F, \delta>$

where

Σ *is the input alphabet*
Q *is a finite set of states*
q_0 *is the initial state,* $q_0 \in Q$
F *is the final state,* $F \subseteq Q$
δ *is a function of transition,* $\delta: Q \times \Sigma \rightarrow Q$

To determine whether a string or sequence belongs to the regular language accepted by the FSA, the automata reads the string from left to right, comparing each one of the symbols of the sequence with the symbols tagging the transitions. If the transition is tagged with the same symbol as the input chain, the automata moves on to the following state, until the sequence is recognized in its entirety by reaching the final state.

Otherwise, a FST is just like a FSA, except that the transitions have both an *input* label and an *output* label. A FST transforms one string into another string if there is a path through the FST that allows it to trace the first string using *input* labels and, simultaneously, the second string using *output* labels. The transition function is tagged with a pair of symbols, which proceed respectively from an input alphabet and an output alphabet. This mechanism can be represented in the form of *finite-state graphs* or transition diagrams, or else as a *matrix* or *transition table*. The transducers can be characterized as directed graphs, whose vertices denote states, while the transitions form the edges, or arcs, with arrows pointing from the initial state to the final state (Figure 1). The FST accepts input strings and associates them with output strings. Formally, a FST is referred to as a 6-tuple (Roche & Schabes, 1995) expressed as shown below:

$FST = <\Sigma_1, \Sigma_2, Q, q_0, F, E>$

where

Σ_1 *is the input alphabet*
Σ_2 *is the output alphabet*

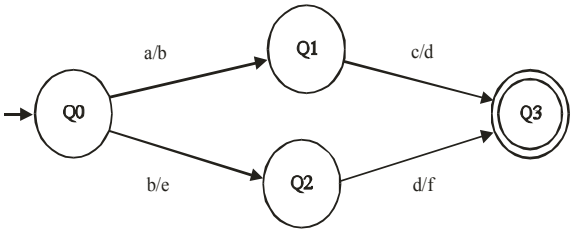
Q is a finite set of states
q_0 is the initial state, $q_0 \in Q$
F is the final state, $F \subseteq Q$
E is a the number of transition relations, $E \subseteq Q \times \Sigma_1 \times \Sigma_2$

One application of FST is to establish a relation between input strings and output strings, that is, between term variants and standardized forms. The objective of this chapter is to defend the application of finite-state techniques for the standardization and grouping of the different variants into an equivalence class that would be configured as the standard form.

STANDARDIZATION OF UNITERM VARIANTS

The standardization techniques based on morphological analysis were first presented in a lexical analyzer developed by a group of computational linguists at Xerox, the *Multi-Lingual Theory and Technology Group* (MLTT). The Xerox analyzer is based on the model of two-level morphological analysis proposed by Koskenniemi (1983). The premise behind this model is that all lexical units can be represented as a correspondence between a *lexical form* (or canonical form) and *surface form* (or inflected form). Further computational development of the Koskenniemi model led to the lexical analyzer by Karttunen known as PC-KIMMO (Karttunen, 1983), the more direct forerunner of the Xerox morphological analyzer.

Figure 1. Finite-state transducers (FST)



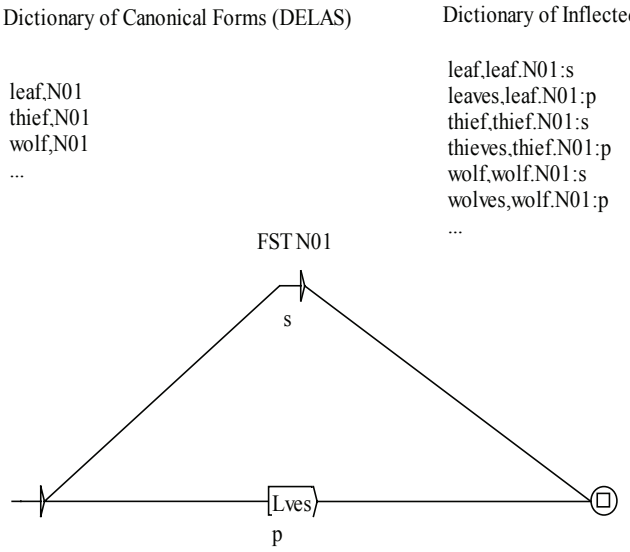
An alternative lexical analyzer based on finite mechanisms is the one proposed by Silberztein (1993), which works without morphological rules. Its technology has been described by Roche and Schabes (1997). A FST associates sets of suffixes to the corresponding inflectional information. In order to produce the inflected forms, one needs to be able to delete characters from the lemma. For this purpose, a delete character operator (*L*) is used, which does not require morphological rules or the help of a finite-state calculus (Silberztein, 1993, 2000). The application developed by Silberztein consists of a dictionary, known as DELAS, of canonical forms with syntactic codes that indicate the POS category of each entry. Each code is linked to a graphic FST made up of an initial node and a final node that describes the path the morphological analyzer should trace. For instance, all the nouns associated with the same inflectional information are associated with the same inflectional FST.

Once the FSTs are compiled, they are projected upon the dictionary of canonical forms, automatically producing the expanded dictionary of inflected forms (known as DELAF) that contains the canonical forms along with inflected forms, POS categories, and inflectional information. With the application of the dictionaries on the lexical units of a corpus, we finally effect two transformations: lemmatization of the inflected forms and POS tagging (Figure 2).

STANDARDIZATION OF MULTITERM VARIANTS

Multiword terms, or NPs, are considered to be more specific indicators of document content than are single words. The identification of phrases using statistical techniques is based on the cooccurrence of the terms, on the application of similarity coefficients and clustering techniques. To identify these, the text must be preprocessed to obtain a phrasal lexicon, defined as a list of NP

Figure 2. The FST N01 associates the sets of suffixes of the DELAS entries to the corresponding inflectional codes (s, singular, and p, plural). In order to obtain the inflected forms from the lemmas in the DELAF entries, the last letter 'f' of the lemma should be eliminated using the delete operator L (Left)



appearing with certain frequency (Fagan, 1989). The subsequent indexing of the documents is based on the identification of the phrases using the lexicon. Salton and McGill (1983) demonstrate that the statistical procedures suffer from certain weaknesses: (1) the selected phrases are very often improperly structured from a syntactic standpoint; and (2) the lack of control in the selection of the phrases may lead to errors that reduce the efficiency of the IRS.

To reduce these problems, we need NLP linguistic methods that can identify the syntactic structures of these constructions and establish some sort of control in the selection of multiword terms. The application of NLP to texts involves a sequence of analytical tasks performed in the separate modules that constitute the linguistic architecture of the system. Among available tools for NP extraction are: the category tagger based on Brill's (1992) rules; the *Xerox morphological analyzer* (Karttunen, Kaplan, & Zaenen, 1992); disambiguation devices of POS categories based on stochastic methods, such as the *hidden Markov model* (HMM) (Cutting, Kupiec, Pedersen, & Sibun, 1992; Kupiec, 1992, 1993); the *NPtool phrase analyzer* (Voutilainen, 1997); or the *AZ noun phraser* (Tolle & Chen, 2000), which combines *tokenizing* with POS tagging (Brill, 1993).

Whether general linguistic resources or specific tools are used, recognizing the variants of phrases continues to be a problem. Ideally, programs would be able to reduce all the variants to normalized forms, where each phrase would be assigned a clearly defined role reflecting the complexity of the syntactic structure. This network of nodes and transitions tagged with POS categories—such as *N* (noun), *AT* (article), *ORD* (ordinal), *CARD* (cardinal), and *DEM* (demonstratives pronouns)—determines sequences in the input, and supplies some form of linguistic information as the output. An entry stream is recognized and transformed into a normalized stream if a path is produced from one node, considered the initial state, to another node, constituting the final state.

To recognize multiword terms through a FST, their structures must be described using regular expressions (RE), defined as a metalanguage for the identification of syntactic patterns. Through this technique, we use the specification of RE to determine the language formed by syntactic patterns. The association of each possible RE with the FSA is represented graphically, with the graphic editor *FSGraph* (Silberztein, 1993, 2000). In order that the FSTs themselves recognize the syntactic patterns, a previous morphological analysis will be needed, giving POS tags to the lexical units. A path between two FST nodes takes place only if the input chain string belongs to the category with which the transition is tagged. In order to use this formalism of the IRS as a means of controlling NP structures, we propose the transformation of the canonical syntactic forms into identifiers of enumerated NP which will be implemented as groupers of structures (Figure 3).

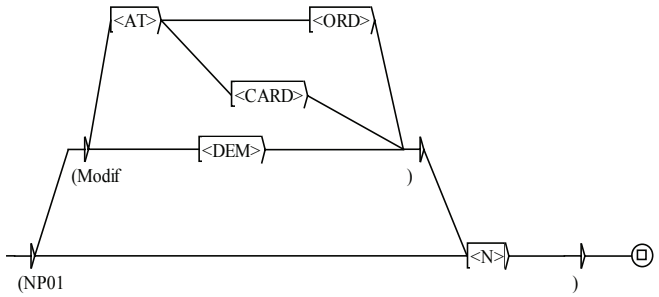
The similar structures can then be transferred to a FST, where the syntactic patterns will be recognized and be standardized into hand-made standardized structures. Thus, we considered that a FST is a method for reducing syntactic structures, comprising two automata that work in a parallel manner. One automata identifies the surface strings, and the other establishes an equivalence relation between the different syntactic structures and an unified structure, or standardized NP:

N = NP01
DEM N = NP01
AT CARD N = NP01
AT ORD N = NP01

FUTURE TRENDS AND CONCLUSION

In IRS, the textual documents are habitually transformed into document representatives by means of linguistic structures configured as indexing terms, classified essentially as single-word terms or uniterms, and multiword terms or multiterms. The single-word terms have morphological variants that refer to the same meaning, and their grouping would improve average recall. Although uniterms may be ambiguous, they usually have relatively few variants, and from a computational treatment, they are easier to formalize. In contrast, multiterms are much more specific, but the grouping of their variants is plagued by difficulties in their identification, because IRS tend to work under the assumption that similar

Figure 3. Relation between variants of syntactic patterns and normalized NP



Regular Relation

<N> <DEM><N> <AT><CARD><N> <AT><ORD><N> => (NP01)

syntactic structures have similar meanings, and should be treated as equivalents, and this is very difficult to regulate in view of the variability of syntactic structures.

There are morphological, lexical, and syntactic variants that cannot be recognized other than through standardization processes of terms. The standardization methods most widely evaluated on retrieval performance involve stemming, segmentation rules, assessing similarity measures of pairs of terms, and clustering techniques. In the linguistic framework, term standardization methods could be considered *equivalence techniques*, employed to regulate linguistic variants and optimize retrieval performance. The application of NLP tools in IR involves morphological analysis, POS taggers, disambiguation processes, lemmatization, and shallow parsing for syntactic pattern-matching. Again we must insist on the influence of language on the results of term standardization. The complexity of terms varies along with the inflectional structure of a language. One interesting study about the morphological phenomena in IRS can be found detailed by Pirkola (2001). Roughly speaking, *synthetic languages*, including French, Spanish, Italian, and the other Romance languages, require term inflection to indicate term function in the sentence. Yet *analytic languages* such as English and German rely on the placement or the combination of terms to indicate their function in the sentence. The synthetic languages have many morphologic variants of single-word terms, whereas the analytic languages have many syntactic variants of multiword terms. Further study should help clarify the positive and negative end effects of these factors on retrieval effectiveness.

To conclude, data quality and standardization are complex concepts governed by multiple dimensions on many variables. Now-a-days, electronic data are at the core of all kinds of treatments; the standardization of terms based on finite-state techniques would adapt well to many other specific applications, such as digital information extraction, bibliometrics, and bioinformatics. The development of finite-state methods may also, however, afford advantages that have not yet been properly explored and evaluated, and their alternative or complementary use might enhance the management of term variants in retrieval performance.

REFERENCES

Abney, S. (1991). Parsing by chunks. In R. Berwick, S. Abney, & C. Tenny (Eds.), *Principle-based parsing*. Dordrecht: Kluwer Academic Publishers.

Arampatzis, A. T., Tsoris, T., Koster, C. H. A., & Van der Weide, P. (1998). Phrase-based information retrieval. *Information Processing & Management*, *34*(6), 693-707.

Brill, E. (1992). *A simple rule based part-of-speech tagger*. Paper presented at the Third Conference on Applied Natural Language Proceedings (pp. 152-155). ACM Press.

Chomsky, N. (1957). *Syntactic structures*. The Hague: Mouton.

Church, K. (1988). A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the Second Conference on Applied Natural Language Processing* (pp. 136-143). Austin, TX: ACL.

Croft, W. B., Turtle, H. R., & Lewis, D. D. (1991). The use of phrases and structured queries in information retrieval. In *Proceedings of the SIGIR 1991*.

Cutting, D., Kupiec, J., Pedersen, J., & Sibun, P. (1992). *A practical part-of-speech tagger*. Paper presented at the Third Conference on Applied Natural Language Processing (pp. 133-140). ACM Press.

Fagan, J. L. (1989). The effectiveness of a non-syntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science*, *40*(2), 115-132.

Frakes, W. B. (1992), Stemming algorithms. In W. B. Frakes & R. Baeza-Yates (Eds.), *Information retrieval: Data structures and algorithms* (pp.131-161). Englewood Cliffs, NJ: Prentice-Hall.

Galvez, C., Moya-Anegón, F., & Solana, V. H. (2005). Term conflation methods in information retrieval: Non-linguistic and linguistic approaches. *Journal of Documentation*, *61*(4), 520-547.

Harris, Z. S. (1951). *Methods in structural linguistics*. Chicago: University of Chicago Press.

Hull, D. A. (1996). Stemming algorithms: A case study for detailed evaluation. *Journal of the American Society for Information Science*, *47*(1), 70-84.

Karttunen, L. (1983). KIMMO: A general morphological processor. *Texas Linguistics Forum*, *22*, 217-228.

Karttunen, L., Kaplan, R. M., & Zaenen, A. (1992). Two-level morphology with composition. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING'92)* (pp. 141-148). ACM Press.

Koskenniemi, K. (1983). *Two-level morphology: A general computational model for word-form recognition and production*. Helsinki: Department of General Linguistics, University of Helsinki.

Kupiec, J. (1992). Robust part-of-speech tagging using a Hidden Markov model. *Computer Speech and Language*, *6*, 225-242.

Kupiec, J. (1993). Murax: A robust linguistic approach for question answer using an on-line encyclopedia. In R. Korfhage, E. Rasmussen, & P. Willett (Eds.), *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 160-169). ACM Press.

Lovins, J. B. (1968). Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics*, *11*, 22-31.

Paice, C. D. (1996). A method for evaluation of stemming algorithms based on error counting. *Journal of the American Society for Information Science*, *47*(8), 632-649.

Pirkola, A. (2001). Morphological typology of languages for IR. *Journal of Documentation*, *57*(3), 330-348.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, *14*, 130-137.

Roche, E. (1996). Finite-state transducers: Parsing free and frozen sentences. In *Proceedings of the ECAI 96 Workshop Extended Finite State Models of Language* (pp. 52-57). Budapest, Hungary: ECAI.

Roche, E., & Schabes, Y. (1995). Deterministic part-of-speech tagging with finite state transducers. *Computational Linguistics*, *21*(2), 227-253.

Roche, E., & Schabes, Y. (1997). *Finite state language processing*. Cambridge, MA: MIT Press.

Salton, G. (1989). *Automatic text processing: The transformation, analysis and retrieval of information by computer*. Reading, MA: Addison-Wesley.

Salton, G., & McGill, M. J. (1983). *Introduction to modern information retrieval*. New York: McGraw-Hill.

Schwarz, C. (1990). Automatic syntactic analysis of free text. *Journal of the American Society for Information Science*, *41*(6), 408-417.

Sheridan, P., & Smeaton, A. F. (1992). The application of morpho-syntactic language processing to effective phrase matching. *Information Processing & Management*, *28*(3), 349-369.

Silberztein, M. (1993). *Dictionnaires électroniques et analyse automatique de textes: Le systčme INTEX.* Paris: Masson.

Silberztein, M. (2000). INTEX: An FST toolbox. *Theoretical Computer Science*, *231*(1), 33-46.

Smadja, F. (1993). Retrieving collocations from text: XTRACT. *Computational Linguistics*, *19*(1), 143-177.

Sparck Jones, K., & Tait, J. I. (1984). Automatic search term variant generation. *Journal of Documentation*, *40*(1), 50-66.

Strzalkowski, T. (1996). Natural language information retrieval. *Information Processing & Management*, *31*(3), 397-417.

Strzalkowski, T., Lin, F., Wang, J., & Pérez-Carballo, J. (1999). Evaluating natural language processing techniques in information retrieval: A TREC perspective. In T. Strzalkowski (Ed.), *Natural language information retrieval* (pp. 113-145). Dordrecht: Kluwer Academic Publishers.

Tolle, K. M. & Chen, H. (2000). Comparing noun phrasing techniques for use with medical digital library tools. *Journal of the American Society for Information Science*, *51*(4), 352-370.

Tzoukermann, E., Klavans, J. L., & Jacquemin, C. (1997). Effective use of natural language processing techniques for automatic conflation of multiword terms: The role of derivational morphology, part of speech tagging, and shallow parsing. In *Proceedings 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'97),* Philadelphia (pp. 148-155).

Voutilainen, A. (1997). *A short introduction to NPtool.* Retrieved August 16, 2008, from http://www.lingsoft.fi/doc/nptool/intro/

KEY TERMS

Finite-State Automata (FSA): A finite-state machine, or finite-state automata, is a mathematical model defined as a finite set of states and a set of transitions from state to state that occur on input symbols chosen from an alphabet.

Lemmatization: Algorithms for reducing a family of words to the same lemma, defined as the combination of the stem and its part-of-speech (POS) tag. This process involves linguistic techniques, such as morphological analysis through regular relations compiled in finite-state transducers.

N-Gram: A *n-gram* is a substring of a word, where *n* is the number of characters in the substring, typical values for n being bigrams (*n=2*) or trigrams (*n=3*).

Noun Phrase (NP): In grammatical theory, a noun phrase is a phrase whose head is a noun, accompanied by a set of modifiers, such as articles, demostratives, quantifiers, numeral, or adjectives.

Stemming: Algorithms for reducing a family of words to a common root, or stem, defined as the base form of a word from which inflected forms are derived. Stemming algorithms eliminate all affixes and give good results for the conflation and normalization of uniterm variants. Within this group, the most effective are the longest match algorithms.

Term Conflation: The process of matching and grouping together variants of the same term that are equivalent. A variant is defined as a text occurrence that is conceptually related to an original term and can be used to search for information in text databases. This is done by means of computational procedures known as conflation algorithms, whose primary goal is the standardization of uniterms and multiterms.