Stemming Methodologies Over Individual Query Words for an Arabic Information Retrieval System

Hani Abu-Salem* and Mahmoud Al-Omari
Department of Computer Science, Mu'tah University, P.O. Box (7), Mu'tah, Karak, Jordan.
E-mail: abusalem@cs.mutah.edu.jo; omari@cs.mutah.edu.jo

Martha W. Evens
Computer Science Department, Illinois Institute of Technology, 10 W. 31 Street, Chicago, IL 60616. E-mail: mwe@math.nwu.edu

Stemming is one of the most important factors that affect the performance of information retrieval systems. This article investigates how to improve the performance of an Arabic Information Retrieval System (Arabic-IRS) by imposing the retrieval method over individual words of a query depending on the importance of the WORD, the STEM, or the ROOT of the query terms in the database. This method, called Mixed Stemming, computes term importance using a weighting scheme that uses the Term Frequency (TF) and the Inverse Document-Frequency (IDF), called TFxIDF. An extended version of the Arabic-IRS system is designed, implemented, and evaluated to reduce the number of irrelevant documents retrieved. The results of the experiment suggest that the proposed method outperforms the Word index method using the Binary scheme and the Word index method using the TFxIDF weighting scheme. It also outperforms the Stem index method using the Binary weighting scheme but does not outperform the Stem index method using the TFxIDF weighting scheme, and again it outperforms the Root index method using the Binary weighting scheme but does not outperform the Root index method using the TFxIDF weighting scheme.

Introduction

Information retrieval (IR) has a strong scientific tradition, and much of its theory is well grounded in experimentation (Van Rijsbergen, 1989). Traditional keyword based retrieval systems have been the subject of experiments for over 35 years. Acceptable results can be produced using simple keyword matching as a basis for retrieval. Furthermore, the addition of ranking techniques based on the frequency of a given matching term within a document collection and/or within a given document makes a considerable improvement to the performance (Salton & McGill, 1983; Sparck-Jones, 1972).

The designers of IRS are concerned with the enhancement of retrieval effectiveness. Relational thesauri and the use of word stemming are widely used methods for this purpose. Stemming is usually done by removing any attached suffixes, prefixes, and/or infixes from index terms before the assignment of the term. Since the stem of a term represents a broader concept than the original term, the stemming process eventually increases the number of retrieved documents. Stemming may also lead to some storage saving (Lovins, 1968; Porter, 1980; Lennon, Peirce, Tarry, & Waillet, 1981; Salton, 1971). In English the stemming processes are commonly carried out by using relatively simple suffix removal algorithms together with special rules to take care of exceptions (Lovins, 1968; Tras, 1976; Porter, 1980). Systems in which word stems are used as index terms as in English and Arabic have, in some cases, shown improvement in retrieval effectiveness (Abu-Salem, 1992; Salton &Lesk, 1968; Salton, 1971); however, Harman's (1991) results show less improvement. Many of the stemming experiments listed in Frakes (1992, Table 8.1, p. 148) failed to find much difference in terms of precision and recall.

The Arabic language is somewhat difficult to deal with due to its right-to-left orientation, the diacritization of scripts, vowels which may or may not be included, and its morphological structure. The grammatical system of the Arabic language is based on a root-and-pattern structure and considered as a root-based language with not more than 10,000 roots (Ali, 1988). The *root* is the bare verb form. The root can be triliteral, which is the overwhelming majority of words, and to a lesser extent, quadliteral, pentaliteral, or hexaliteral, each of which generates increased verb forms

* To whom all correspondence should be addressed.

Received October 9, 1997; revised June 5, 1998; accepted September 2, 1998.

and noun forms by the addition of derivational affixes (Saliba & Al-Dannan, 1989). A *stem* is a combination of a root and derivational morphemes to which an affix (or more) can be added (Gleason, 1970). When applying this definition to Arabic, the verb roots and their verb and noun derivatives are considered as stems. *Affixes* are prefixes, suffixes, and infixes (morphemes) attached at the beginning, at the end, and in the middle of the root respectively. For example, the Arabic word الحاسبات (alhasebat) consists of the following elements: the prefixes ال (al), the stem حاسب (haseb), and the suffix ات (at). Notice that the stem is derived from the triliteral verb حسب (hasab).

Arabic-IRS is a microcomputer-based Arabic information retrieval system, which was developed by Abu-Salem (1992). The system was used for a series of experiments in information retrieval involving index terms and query enhancement with relational thesauri. Arabic-IRS was used in experiments with the use of words, stems, or roots as index terms using 32 boolean queries against a database of 120 documents with categories, titles, authors, sources, abstracts, etc., in which a binary weighting scheme was used. The results of these experiments proved that this system functions better with roots than with stems as index terms and better with both roots and stems than with words.

Al-Kharashi (1991) and Al-Kharashi and Evens (1994) built an experimental Microcomputer Arabic Information Retrieval System, Micro-AIRS. They studied the effect of using words, stems, and roots as index terms on the effectiveness of the retrieval of Arabic bibliographic records using 355 records of document titles without abstracts as a database. In this experiment, the best results came when roots were used as index terms. The use of document titles alone proved to be less effective for content analysis purposes than the use of full-document abstracts (Abu-Salem, 1992).

Abu-Salem and Omari (1995) developed an extended version of their system (Arabic-IRS) to investigate how the Inverse-Document Frequency Weighting function affects the retrieval performance of documents that have abstracts by comparing three different indexing methods. These methods involve the use of words, stems, and roots as index terms. The experimental results revealed that the stems were superior index terms compared to words. The roots retrieval method performed significantly better than the word retrieval method and performed significantly better than the stem retrieval method at higher recall levels.

The present work concerns a test of the hypothesis that the best retrieval performance will result from applying different stemming methods to different query terms according to their individual properties. The stemming method, applied to a certain query term, depends on the average term importance of its word, its stem, or its root in the database. The goal is to develop a complete on-line retrieval system that interprets queries using different stemming and indexing techniques to retrieve relevant abstracts with reasonable response time.

An overview of different weighting schemes used to measure term importance will be given in the next section. The proposed stemming method, the results of the experiment, and the conclusions will be presented in sequence.

Term Weighting Schemes

One way to speed up a search for information items is to develop an indexing technique that provides access to segments of a database. Different indexing methods have been used in (Abu-Salem, 1992; Abu-Salem & Omari,1995; Al-Kharashi, 1991; Al-Kharashi & Evens, 1994), with the full-word, the stem of the word, or the root of the word as index terms. The introduction of term weights for index terms and query terms may help to distinguish terms that are more important for retrieval purposes from other less important ones.

The binary weighting scheme (Abu-Salem, 1992; Al-Kharashi & Evens, 1994), means that all allowable index terms are assigned the same weight. The similarity coefficient is one of the most common ways to find the relationship between the queries and the retrieved documents. The normalized Cosine similarity coefficient between query Q and document D using the *Binary* weighting scheme is computed as follows:

$$similarity(Q, D) = \frac{|Q \cap D|}{|Q|^{1/2} \cdot |D|^{1/2}},$$

where, $|Q|$ = number of terms in Q, $|D|$ = number of terms in D, and $|Q \cap D|$ = number of terms appearing jointly in Q and D.

When all document terms carry weight assignments, it is easy to rank the retrieved documents in decreasing order of their similarity coefficients according to the query, making it easy hence to display them in decreasing order of presumed importance.

One easily computable term weighting system that has consistently given excellent retrieval results is the Inverse Document-Frequency Weighting scheme (*IDF*). Each term weight inversely proportional to the total number of documents to which that term is assigned (Abu-Salem & Omari, 1995; Luhn, 1957, 1958; Salton, 1975; Salton, 1989; Salton & Buckely, 1988; Salton & Wu 1981). This weighting scheme reflects the importance of a term within the document itself and the database as a whole. The inverse document-frequency weight of term k for a given database is computed as follows:

$$IDF_k = \log_2 \frac{N}{D_k} + 1,$$

where, N is the number of documents in the database and D_k is the number of documents in the database that contain one or more instances of term k.

TABLE 1. Average recall-precision using the Word Index Method for the Binary and the *TFxIDF* Weighting Schemes, and Using the Mixed Stemming Method.

Recall	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Mixed stemming method	1.0000	0.9141	0.7635	0.6581	0.5741	0.4488	0.3580	0.2970	0.2508	0.2004
Binary weight (word method)	0.3750	0.2500	0.1406	0.0859	0.0586	0.0449	0.0381	0.0190	0.0095	0.0048
TFxIDF weight (word method)	0.9062	0.7031	0.4922	0.3555	0.2715	0.2295	0.1929	0.1433	0.1185	0.1061

Another weighting scheme that uses the term frequency in individual documents and the Inverse Document-Frequency is called the *TFxIDF*. In this scheme, for each term T_k in document D_i a weighting factor w_{ik} is computed and composed in part of the term frequency TF_{ik} which is the occurrence frequency of term k in document i, and in part of the inverse document-frequency factor IDF_k for the term:

$$w_{ik} = TF_{ik} \cdot IDF_k.$$

Each document D_i is represented by the vector of weighted terms. For each term T_k in query Q, the weighting factor y_k is computed using the formula $y_k = IDF_k$. Because terms assigned to queries are fewer than to documents, and the occurrence frequency of the query terms do not exceed 1, the term-frequency component of the query-term weights can be eliminated. Given query- and document-term vectors of the form

$$Q = (y_1, y_2, \ldots, y_t) \text{ and}$$

$$D_i = (w_{i1}, w_{i2}, \ldots, w_{it})$$

where t is the total number of distinct terms in the collection, and the y_j represent the inverse document-frequency weights, and the w_{ij} are defined as combined term-frequency and *IDF* weights. The following Cosine similarity coefficient can be used to compute the query-document similarity value between boolean query Q and document D_i:

$$SIM(Q, D_i) = \frac{\sum_{k=1}^{t} y_k w_{ik}}{\sqrt{\sum_{k=1}^{t} y_k^2 \cdot \sum_{k=1}^{t} w_{ik}^2}}.$$

Mixed Stemming Method

Automatic stemming algorithms in the Arabic language are still very expensive and complicated due to the nature of the language itself in which the root is mostly the triliteral verb with many prefixes, infixes, and suffixes. In English, however, several stemming algorithms have been widely used to enhance the retrieval behavior in different information retrieval experiments. Some of these experiments have shown improvement in retrieval performance (Frakes, 1992), but others did not (Harman, 1991). Harman showed (1991) that individual queries were affected by stemming, but the number of queries with improved performance was close to the number with poorer performance. The result is little change for the overall test collection examined.

The effort to improve the Arabic-IRS performance is based on the hypothesis that indexing the term variants of

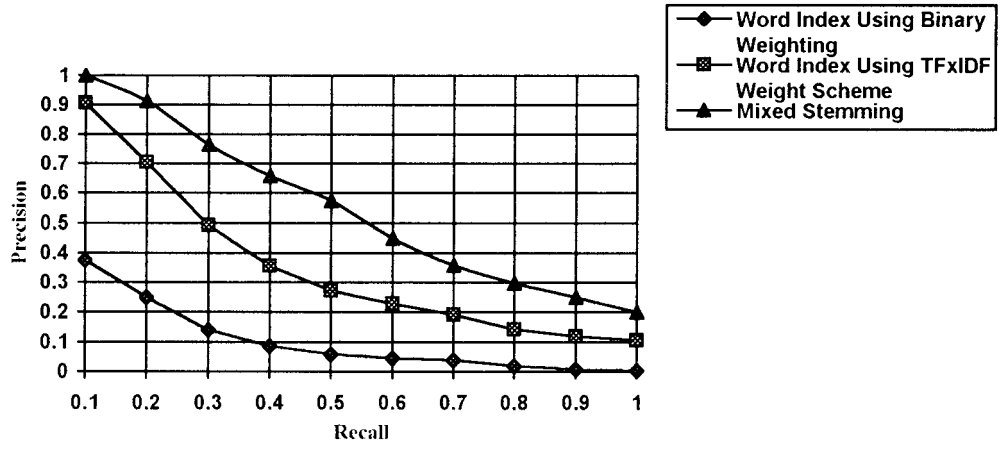

FIG. 1. Average recall–precision using Word index methods.

TABLE 2. Average Recall-Precision Using the Stem Index Method for the Binary and the *TFxIDF* Weighting Schemes, and Using the Mixed Stemming Method.

Recall	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Mixed stemming method	1.0000	0.9141	0.7635	0.6581	0.5741	0.4488	0.3580	0.2970	0.2508	0.2004
Binary weight (stem method)	0.7960	0.6687	0.4554	0.3342	0.2736	0.1993	0.1465	0.1045	0.0678	0.0496
TFxIDF weight (stem method)	1.0000	0.9809	0.9226	0.8518	0.7877	0.7153	0.6189	0.5083	0.3837	0.2856

words that are already widespread in the database degrades performance (Abu-Salem, 1992; Abu-Salem & Omari, 1995). A new technique is proposed here to improve the performance of Arabic-IRS by choosing an automatic index method according to the average importance of the word, the stem, or the root of the individual query terms in the document collection. Term importance is measured using the *TFxIDF* weighting scheme.

The experiment was performed on a database of 120 Arabic documents with abstracts, and a total of 32 queries were generated by computer science experts. An Arabic dictionary was built manually which contains the isolated terms from the database, excluding the stop words, with two more fields, the stem of the word and the root of the word. For each of the three fields (word, stem, and root) an additional field is added. These additional fields contain the average of the word weights in all documents, the average of the stem weights in all documents, and the average of the root weights in all documents.

For each term of the query, if found in the dictionary, the index method used for retrieval is decided according to the largest average weight of that term according to its word, its stem, or its root in the dictionary. For example, the search argument "$Term_A$ AND ($Term_B$ OR $Term_C$)" might be expressed automatically by the system as "STEM:$Term_A$ AND (WORD:$Term_B$ OR ROOT:$Term_C$)" . In this example the system has chosen the STEM index method for $Term_A$ because the average weight of the stem is larger than either the average weight of the word or the root. The system has chosen the WORD index method for $Term_B$ because the average weight of $Term_B$ is larger than either the average weight of the stem or the root. The system has similarly chosen the ROOT index method for $Term_C$.

The similarity coefficient values are computed between the query and the retrieved documents. The retrieved documents are then presented to the user in decreasing order of their similarities.

Results and Discussion

Evaluation of the retrieval system plays an important role in judging the efficiency and effectiveness of the retrieval process. Several evaluation criteria were used in different experiments, among them, recall and precision. *Recall* is defined as the ratio of the number of relevant documents that are retrieved to the total number of relevant documents. *Precision* is defined as the ratio of the number of relevant documents that are retrieved to the total number of retrieved documents (Salton & McGill, 1983). This experiment was carried out on a database of 120 documents in the field of Computer Science and a total of 32 queries generated by user experts, whose judgments regarding the relevance of each document to each query were taken into consideration. The similarity coefficients are calculated between each query and its corresponding set of retrieved documents. Recall and precision are measured after the documents are presented to the user in decreasing order of their similarity coefficient values. This produces a sequence of recall–

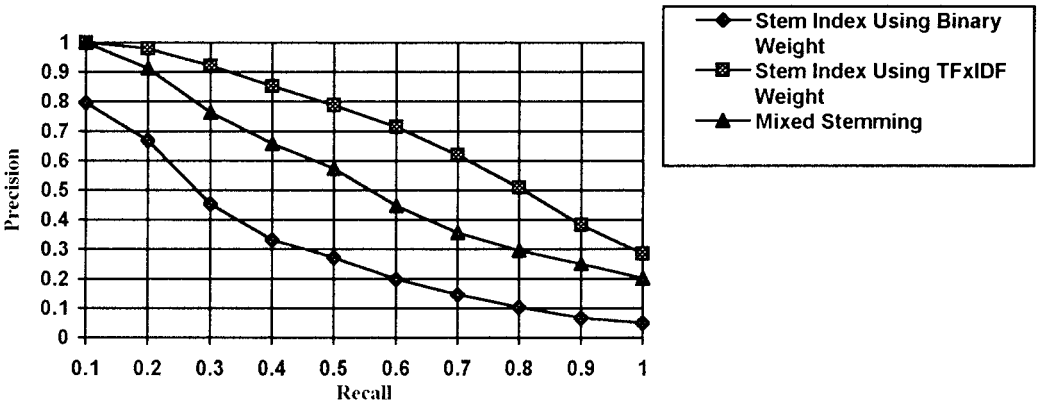

FIG. 2. Average recall–precision using the Stem index method.

TABLE 3. Average Recall-Precision Using the Root Index Method for the Binary and the *TFxIDF* Weighting Schemes, and Using the Mixed Stemming Method.

Recall	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Mixed stemming method	1.0000	0.9141	0.7635	0.6581	0.5741	0.4488	0.3580	0.2970	0.2508	0.2004
Binary weight (root method)	0.8400	0.7146	0.5466	0.4008	0.3174	0.2599	0.1973	0.1428	0.0870	0.0591
TFxIDF weight (root method)	1.0000	0.9797	0.9401	0.8641	0.8532	0.8199	0.7873	0.7808	0.7469	0.6398

precision pairs that can be plotted as a curve (Salton & McGill, 1983). In such a graph, for each recall point there is a corresponding precision value.

The average precision values at 10 recall points for all queries using the word indexing method for both the Binary and the *TFxIDF* weighting schemes and the Mixed Stemming method are shown in Table 1 and Figure 1. The summaries provided by the average recall–precision values suggest that the proposed Mixed Stemming method of the individual query words outperforms the Word index method using the Binary and the *TFxIDF* weighting schemes.

The average precision values at 10 recall points for all queries using the Stem index method for both the Binary and the *TFxIDF* weighting schemes and the Mixed Stemming method are shown in Table 2 and Figure 2. The summaries provided by the average recall–precision values suggest that the proposed Mixed Stemming method of the individual query words outperforms the Stem index method using the Binary scheme but does not outperform the Stem index method using the *TFxIDF* weighting scheme.

The average precision values at 10 recall points for all queries using the Root index method for both the Binary and the *TFxIDF* weighting schemes and the Mixed Stemming method are shown in Table 3 and Figure 3. The summaries provided by the average recall–precision values suggest that the proposed Mixed Stemming method of the individual query words outperforms the Root index method using the Binary scheme but does not outperform the Root index method using the *TFxIDF* weighting scheme.

Conclusions

Arabic-IRS was designed as an experimental system to investigate indexing and retrieval processes with relational thesauri for Arabic bibliographic data. The system used the Binary Weighting Scheme. It allows the user to use only one type of index method at any given time, but the extended version developed in this research, has the ability to impose the retrieval method over individual terms of a query depending on the importance of the word, the stem, or the root of the query terms in the collection. The results show that the Mixed Stemming method outperforms all three (word, stem, and root) index methods using the Binary weighting scheme and the Word index method using the *TFxIDF* weighting scheme. But the Mixed Stemming method did not outperform the stem and the root indexing methods using the *TFxIDF* weighting scheme.

Although it was not a direct aim of this experiment, the Root index method using the *TFxIDF* weighting scheme is shown to be the best of all the seven methods compared. The benefits of using stems or roots as index terms for Arabic are clearer than for English. This is due to the fact that the Arabic language is a root based

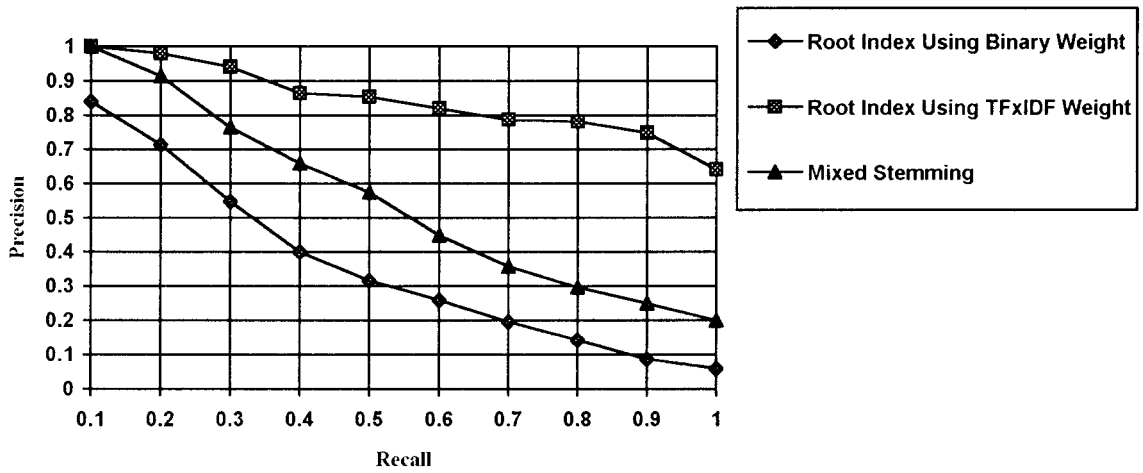

FIG. 3. Average recall–precision using the Root index method.

TABLE 4. Average Precision at 50% Recall for the Seven Runs.

Method	Precision
Root—*TFxIDF*	0.8532
Stem—*TFxIDF*	0.7877
Mixed method	0.5741
Root—Binary	0.3174
Stem—Binary	0.2736
Word—*TFxIDF*	0.2715
Word—Binary	0.0586

language where each root can generate an increasing number of words.

Finally, the results for the seven runs are summarized, in rank order of precision at 50% recall, in Table 4.

Acknowledgments

This work was partially supported by the Italian Ministry of Foreign Affairs under a Cooperation Project between Italy and Jordan.

References

Abu-Salem, H. (1992). A microcomputer based Arabic bibliographic information retrieval system with relational thesauri. Ph.D. dissertation, Illinois Institute of Technology, University Microfilm, Ann Arbor, MI.

Abu-Salem, H., & Omari, M. (1995). Comparing words, stems, and roots as index terms using term weighting scheme in an Arabic information retrieval system. Proceedings of ICECS 95, Amman, Jordan.

Ali, N. (1988). Computers and the Arabic language. Cairo, Egypt: Al-Khat Publishing Press, Ta'reep.

Al-Kharashi, I. (1991). Microcomputer based Arabic information retrieval system, comparing words, stems, and roots as index terms. Ph.D. dissertation, Illinois Institute of Technology, University Microfilm, Ann Arbor, MI.

Al-Kharashi, I., & Evens, M. (1994). Comparing words, stems, and roots as index terms in an Arabic information retrieval system. Journal of the American for Information Science, 45(8), 548–560.

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

Gleason, H.A. (1970). An introduction to descriptive linguistics (3rd Edition, p. 69). New York, NY: Holt, Rinehart and Winston.

Harman, D. (1991). How effective is suffixing? Journal of the American Society for Information Science, 42(1), 7–15.

Lennon, M., Peirce, D., Tarry, B., & Waillett, P. (1981). An evaluation of some conflation algorithms for information retrieval. Journal of Information Science, 32, 177–188.

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

Luhn, H.P. (1957). A statistical approach to mechanized encoding and searching of literary information. IBM Journal of Research and Development, 1, 309–317.

Luhn, H.P. (1958). The automatic creation of literature abstracts. IBM Journal of Research and Development, 2, 159–165.

Porter, M.F. (1980). An algorithm for suffix stripping, program, 14, 130–137.

Saliba, B., & Al-Dannan, A. (1989). Automatic morphological analysis of Arabic: A study on content word analysis. Proceeding of the first Kuwait computer conference (pp. 3–8).

Salton, G. (1975). A theory of indexing, regional conference series in applied mathematics. Society for Industrial and Applied Mathematics (No. 18), Philadelphia, PA.

Salton, G. (1989). Automatic text processing. Reading, Massachusetts: Addison-Wesley.

Salton, G. (1971). The SMART Retrieval System—Experiments in Automatic Document Processing. Englewood Cliffs, NJ: Prentice-Hall Inc.

Salton, G., & Buckely, C. (1988). Term weighting approaches in automatic text retrieval. Information Processing and Management, 24(5), 513–523.

Salton, G., & Lesk, M.E. (1968). Computer evaluation of indexing and text processing. Journal of the ACM, 20, 258–278.

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

Salton, G., & Wu, H. (1981). A comparison of search term weighting: Term relevance vs. inverse document frequency. Proceeding of the Fourth International Conference on Information Storage and Retrieval, Oakland, California, May 31–June 2 (pp. 30–39).

Sparck-Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval, Journal of Documentation, 28, 11–20.

Tras, A. (1976). Stemming as a system design consideration. ACM SIGIR Forum, 11, 9–16.

Van Rijsbergen, C.J. (1989). Towards an information logic. ACM SIGIR: Proceeding of the 12th International Conference on Research and Development in Information Retrieval (pp. 77–86).