Article

JIS

Journal of Information Science
2014, Vol. 40(6) 736–748

DOI: 10.1177/0165551514540400
jis.sagepub.com
SAGE

Text messaging and retrieval techniques for a mobile health information system

Ademola O. Adesina
Department of Computer Science, University of the Western Cape, South Africa

Kehinde K. Agbele
Department of Computer Science, University of the Western Cape, South Africa

Ademola P. Abidoye
Department of Computer Science, University of the Western Cape, South Africa

Henry O. Nyongesa
Department of Computer Science, University of the Western Cape, South Africa

Abstract
Mobile phones have been identified as one of the technologies that can be used to overcome the challenges of information dissemination regarding serious diseases. Short message services, a much used function of cell phones, for example, can be turned into a major tool for accessing databases. This paper focuses on the design and development of a short message services-based information access algorithm to carefully screen information on human immunodeficiency virus/acquired immune deficiency syndrome within the context of a frequently asked questions system. However, automating the short message services-based information search and retrieval poses significant challenges because of the inherent noise in its communications. The developed algorithm was used to retrieve the best-ranked question–answer pair. Results were evaluated using three metrics: average precision, recall and computational time. The retrieval efficacy was measured and it was confirmed that there was a significant improvement in the results of the proposed algorithm when compared with similar retrieval algorithms.

Keywords
frequently asked question (FAQ); information retrieval (IR); mobile health (mHealth); mobile Health Information system; question and answer (Q&A) system; short message service (SMS)/text message

1. Introduction

The short message service (SMS)-based information retrieval is a way of accessing information necessitated by the rapid development of mobile telecommunication. The technology is characterized by instant access to information as a response to SMS enquiries. The SMS-based information request is considered unique because of the restricted size available for the reply, so only a few results can be returned for any given query. A mobile retrieval search system enables the user to obtain extremely concise and appropriate responses from queries across arbitrary topics. Users may be forced to rephrase or reformulate the query if their answers are not made available in the preliminary pages of the search response. Unfortunately, there is a limit to what a mobile phone user can download compared with what is downloadable from a desktop system or microcomputer. Mobile search users rarely employ the advanced search feature of the search engine but prefer to expend extra energy in reformulating the query [1, 2].

Corresponding author:
Ademola O. Adesina, Department of Computer Science, University of the Western Cape, Private Bag X17 Bellville 7535, South Africa.
Email: inadesina@gmail.com

In the meantime, advances in mobile communication have generated the concept of text messaging. This computer-mediated communication has its own peculiarities, whereby users have their own patterns of writing, inventing new abbreviations and using non-standard orthographic forms [3]. This language is not in alignment with the pattern of traditional natural languages because of the restriction in the number of characters permissible to be written on the mobile device, bandwidth of digital communication and limited memory capacity of the mobile phone [4, 5]. The SMS communication provides a platform where messages can be delivered even when the recipient is engaged in voice communication or is otherwise unable to attend to a call. The text message has created a social network environment for peer groups where information is shared. For example, in mobile health (mHealth) technology, SMS has played a significant role in bridging the gap in communication not only between the patients and health workers [6–8] but also between healthcare workers and medical resources, for example, the medical library [9].

SMS communication is used for information access and retrieval in healthcare-related applications. Its introduction minimizes visits of physicians to patients, and is useful for drug prescriptions, consultancy services, appointment reminders, health and prevention reports, bills and other forms of information. This leads to better understanding and education about healthcare issues, and in turn, reduces the cost of healthcare provision [10–13]. For instance, in South Africa, SMS is used to remind tuberculosis patients to take their drugs. The tuberculosis drug *Rifafol* needs to be taken daily and on a consistent basis to be effective. SMS texts which are written in English and local languages – Afrikaans and Xhosa – are sent at a pre-determined time daily to the patient. This is done for a period of six months for a complete treatment [14]. In addition, in 2008, *SIMPill* was implemented in South Africa to remind tuberculosis patients of their medication. A Subscriber Identity Module (SIM) card is placed on the bottle top, which sends an SMS text every time the bottle is opened. A reminder in the form of an SMS text is sent to the patient or relative if the bottle is not opened at the expected time [15].

Generally, an information retrieval process begins when a user enters an enquiry into the search engine with the expectation of getting reasonable feedback. An enquiry is a formal statement of the user's information needs expressed in a formal language [16]. The feedback represents the list of sensible answers to the enquiry made by the information seeker. This list is made available as a result of a comparison between the keyword terms of the query statement [17, 18] and the repository of the answers in a database [19, 20]. Natural language and social network communication (SNC) languages can be used as query statements to source information on the search engines. SMS as an example of SNC is a preferred form of communication for many youths [21–23]. For this communication paradigm, building automated question-answering systems has proved difficult because of the various means the users have employed of representing formal language. For example, in the datasets collected for the experiment, *tomorrow* as an English word is expressed in more than 20 SMS versions – *tomoz, tomorro, tomorrw, tomora, morrow, mora, tom, 2mora, tomoro, 2morrow, tmw, 2mrow, 2morow, 2morro, 2mrrw, 2moz, 2mrw, amoro, tomorrrow, 2moro, tmrrw* and *tomrw*. The translation of SMS variants into the Standard English form (*tomorrow*) is of utmost importance in SMS normalization. This is the stage of correcting the erroneous forms in which the SMS text appears (SMS normalization is, however, beyond the scope of this paper).The freedom of SMS writing poses a great challenge to its normalization. Other natural language processing techniques like stemming, regression, classification, clustering and stop word identification are almost impossible until there is a correction of the noisy form of the SMS.

Using a Google search as a benchmark, the typical results of an SMS-based search can be considered using a query sentence – *wn d u intt arv thrpy* – extracted from an English query 'when do you initiate antiretroviral therapy?' Google responds only with a normalized form of *thrpy* as *therapy* and translates the abbreviation *arv* to *antiretroviral*. This is a usual experience for SMS information seekers. Google appears to be the best web search engine in terms of average precision and response time [24]. When used, the SMS query results mostly take the form of Garbage In Garbage Out, and as such are not helpful to the SMS user. Normally, when a user mistypes an input query, the system will suggest an alternative query sentence, in order to continue the semantic-based search [25]. Sometimes, suggestions made by the search engine are far from the intent of the SMS user, for example, in the search that was performed, *wn d u* were joined together as *wndu*.

There is a need to *clean/normalize* the SMS in order for it to play a role in question answering (QA) systems [20, 26–29]. An SMS-based QA retrieval system accesses information in the form of questions and answers with the use of SMS services on the mobile phone platform. The QA systems may appear in four guises. The first is *natural language processing* – this is a situation whereby users send a query in natural language for enquiries on phones or mobile devices, and the answers are returned in natural language. The Google web search system uses the text of links to index documents [30] and processes the query. For example, using a query sentence 'when do you initiate antiretroviral therapy?' will return a long list of documents about *antiretroviral*, because the search system has found references that include the word *therapy* that most frequently point to documents discussing *antiretroviral therapy*. The second is *human intervention* – messages are sent in the form of natural language to a particular agent. Normally, the agent, who is an expert, gives the

answer to the request. This is mostly common with expert systems, like MYCIN, where enquiries are made so as to determine the kind of ailment and treatment procedure. MYCIN is a computer program designed to provide attending physicians with advice comparable to that which they would otherwise get from a medical consultant. To use MYCIN, the attending physician must sit in front of a computer terminal that is connected to a DEC-20 (one of Digital Equipment Corporation's mainframe computers) where the MYCIN program is stored. When the MYCIN program is evoked, it initiates a dialogue. The physician types answers in response to various questions. Eventually MYCIN provides a diagnosis and a detailed drug therapy recommendation [31]. The third is the *information retrieval method* – the corpus will be searched for a possible answer to the request, and the answer may be delivered after the enquirer has responded to the request from the machine, for instance, to type specific code to retrieve information. This method is common in interactive voice response systems, which are the interfaces that stand in for a live operator or telephone attendant to route a user through a company's telephony system. One might be familiar with such phrases as, 'for English, press 1' or 'Please enter your 9-digit social security number now'. Interactive voice response systems are used for enquiries in some companies for customer support and routine billing system [32, 33]. The fourth is *frequently asked question retrieval* – there is a ready-made answer to every enquiry that may be requested from the user, for example, health-related issues. The database is searched for the enquiry and an appropriately matched answer is returned. The *FAQ FINDER* system is an example of frequently asked question retrieval system that uses a natural language question-based interface to the distributed information sources, specifically files organized as question/answer pairs such as frequently asked question (FAQ) files. In using this system, the user enters question(s) in natural language and the system presents answer(s) to the question, using FAQ files as a resource [34, 35]. In this paper, the focus is on the FAQ retrieval system.

The frequently asked question is transformed to an SMS-based FAQ retrieval system. This is designed to give a set of FAQs for a query written in SMS language. The FAQ may be: (1) *monolingual FAQ retrieval* – the FAQ and SMS datasets are of the same language and the only challenge is to get the best match between the two datasets; (2) *cross-lingual* – the FAQ and SMS datasets are not of the same language, and in this case, the challenge is to get the best match between two dissimilar datasets; or (3) *multilingual* – the FAQ and SMS datasets comprise many languages and the challenge is to get the best match between various languages or datasets. In this paper, the monolingual SMS-based FAQ retrieval system is treated as the question-answering system. The algorithm presented in this paper is on SMS written in English language.

The paper is organized as follows: in the next section, the state-of-the-art method for SMS-based information retrieval system is reviewed. The system flowchart in building the SMS-query system is presented in Section 3. Section 4 discusses the research problems and methodology adopted in building the SMS-based FAQ system. The SMS-based information search and retrieve algorithms of the proposed (*SMSql*) and existing (*tf-idf*) algorithms are described in Section 5. The performance evaluation and the metric indices are discussed in Section 6. In Section 7, the results of the experiment are presented. Finally the paper is concluded in Section 8.

2. Related work

It is crucial that relevant answers are provided for users when the enquiry is made, otherwise this can lead to query abandonment or further iteration of the request [36]. Hence it is important for the search to present the most relevant document in the FAQ collection as the answer to the SMS-based request. Burke et al. [37] used a natural-language-processing question answering system that uses FAQ files as its knowledge base. The technique is based on four assumptions used to convert the *FAQFINDER* system: (1) organizing the FAQ file in QA format; (2) setting the information locality within the QA pair; (3) determining the question's relevance within the QA in order to find the match; and (4) possessing a general knowledge of the languages for question matching. The user's query terms are matched with the FAQ files. The *FAQFINDER* search process was limited to a small set of FAQ files that are likely to have the best match to the user's query.

Mogadala et al. [38] used a language modelling (LM) approach to match *noisy* SMS text with the right FAQ. The team developed a dictionary-based approach for SMS text normalization. The *cleaned* SMS text is then matched with the FAQ using an LM method before the corresponding response to the query is released. Mogadala et al.'s [38] experiments use a combination of SMS datasets of English, Hindi and Malayalam languages with their corresponding FAQs in different combinations for the monolingual task, and FAQs in Hindi and the English language for the cross-lingual task. In both sets of experiments, the percentage of the languages is continuously varied in order to retrieve information from their FAQ databases using English SMS queries. The FAQs are divided into three different collections: (1) the questions only; (2) the answers only; and (3) combinations of questions and answers of the three languages. The results show that developed LM questions outperform both answers and combinations of questions and answers for matching SMS queries. The

LM model does not give consideration to synonyms. It is word-dependent. This means that any other answer that could be chosen in the FAQ answer dataset may not be considered.

Hogan et al. [26] identified SMS-based FAQ retrieval systems as having three steps: (1) SMS normalization; (2) retrieval of ranked results; and (3) identification of out-of-domain query results. In order to normalize the SMS FAQ queries, a set of transformation rules was created and the corpora were manually annotated. The tokens were aligned with the original text messages to give a one-to-one correspondence between the original and corrected tokens. The documents and SMS questions underwent the same pre-processing. In the research of Hogan et al. [26], each SMS token was examined (if it remains unchanged) and then the corrected token was substituted. A set of candidate lists was generated, and the best candidate in the context was selected as the correction. The best candidate was selected using three methods: (1) manually annotated data was used as a correction rule to get the best transformation for the SMS tokens and the frequency of use of the correction rules became a criterion for calculating the normalized weights of the replacement of SMS token in the corpus; (2) candidate corrections were created by consonant skeletons – a consonant skeleton is the withdrawal of vowels from a word, leaving only the 'consonant' of the word, for example, 'medicine → mdcn', 'health → hlth' [39], and the mapping between the consonant skeletons and the words produces additional correction candidates for the query words; and (3) candidates were generated when all words in the corpus were compared with the prefix of the question words, to confirm that there was a truncation. The three methods produced replacement candidate lists, which were merged by adding their weightings from their term frequency. The token scores were calculated using the maximum product of that weight and the *n*-gram score of the corrected token. Hogan et al. [26] used a manual annotation of the dataset, but this may be cumbersome for large corpora. The experiment was performed on monolingual English SMS datasets with different retrieval engines (*Solr, Lucene* and a combination of the two search engines) and approaches. The best result from the candidate list was retrieved by ranking the weighted scores of a list of question–answer pairs. The evaluation of the results involved comparing out-of-domain results when tested on the two search engines. The SMS normalization approach is token-based and all the tokens were processed.

Kothari et al. [19] designed an automatic FAQ-based question answering system. The method involved promoting SMS query similarity to FAQ-questions. This was done through a combinatorial search approach. The search space consisted of combinations of all possible dictionary variations of tokens in the noisy query. The combinatorial search system modelled an SMS query as a syntactic tree matching so as to improve the ranking scheme after candidate words had been identified. Initial processing of noise removal was introduced so as to improve the information retrieval efficiency. The model involved the use of a dictionary, and mapped the SMS query to the questions in the corpus. The noise removal step was, however, computationally expensive [40]. The system developed by Kothari et al. [19] did not involve training SMS data on text normalization. It had the advantage of handling semantic variations in question formulation but the method failed to discuss the choice of homophonic words in the context of automatic speech recognition. Kothari et al. [19] depended on a scoring function for the choice of selecting FAQ questions. Thus, in cases where there is a tie over the score function, it would be difficult to rank the question, and other factors, such as the proximity measurement of the SMS query and FAQ token as proposed by Jain [41] and Joshi [42].

An *n*-gram count-based algorithm developed by Jain [41] took into account various *n*-grams in order to calculate the score of questions from the corpus. This was similar to the approach used to develop *SMSql* (Section 5). The score of different FAQ questions from the candidate sets was then calculated. The maximum score among the set was therefore returned with its corresponding answer in the FAQ database. Two factors were considered that led to an enhancement in evaluating the FAQ score in the candidate set. They were the proximity of the SMS query and FAQ tokens, and a comparison of the question sentence length of the matched tokens from the SMS query to the FAQ questions under consideration [41, 42]. When the algorithm was evaluated on many real-life FAQ datasets from different domains, the results showed significant improvement in terms of the accuracy compared with Kothari et al. [19]. This approach did not give consideration to synonyms, that is, it was word-dependent as answers were chosen only from the FAQ answer dataset.

Chen et al. [4] proposed *SMSFind* as another type of SMS-based information retrieval model. This was designed to deliver the final search response to a normalized SMS query. It used a conventional search engine in its back end to provide an appropriate answer for the SMS request. *SMSFind* used translated SMS queries. Typically, the arrangement contained an SMS term or a collection of consecutive terms in a query that provides a *hint* as to what the user is looking for. The *hint*, provided by the user or automatically generated from the document, was used to address the information extraction problem. *SMSFind* used this *hint* to address the problem as follows: given the top search responses to a query from a search engine, *SMSFind* extracted snippets of text from within the neighbourhood of the *hint* in each response page. *SMSFind* scored snippets and ranked them across a variety of metrics. The *hint* extracted was used to determine the answer to the request. It was scored based on a *top-n* list for each page. The highest score was released as an answer to

the request [4]. The use of *hints* in the algorithm was considered as a supervised learning approach [43, 44] and it was expensive to generate and store. The research never considered the contextual information of the searches and the searching was limited to the constituent of the *hint*.

The research presented shares similarities in the area of application, that is, health-related matter, with the research of Anderson et al. [45] and Masizana-Katongo et al. [46], but the two research groups used SMS parsing technique to query the search engine after the SMS token had been disambiguated using context-free grammar. In our approach, the SMS term was taken as a query while the FAQ was considered as a document for the SMS-based retrieval method. While their research was applied to a multilingual scenario, ours considered the monolingual scenario. The system flowchart of the SMS-based information retrieval system is discussed in the next section.

3. System flowchart of the proposed SMS-based retrieval algorithm (*SMSql*)

The flowchart in Figure 1 shows a typical SMS-based retrieval system. The flowchart is drawn in order to understand, evaluate and design the SMS-based information retrieval system. The SMS terms are sent into the SMS normalization/translation process. The normalized SMS terms are compared with the FAQ-query terms in the FAQ English database query set. The FAQ query terms are statistically selected as the *keywords.*

The FAQ-queries are ranked according to the extent of the similarities of the SMS-queries and FAQ-queries in the *ranked list of pre-defined FAQ queries*. The result of the SMS request is presented to the *texter* based on the user's judgement, that is, is it satisfactory or not? However, there is room for a reformulation process to take place. Dissatisfaction of the users of information leads to disengagement from the search system [47–49]. Expected results and satisfactory answers may not be presented to the user if the FAQ-query dataset does not have the query terms in common.

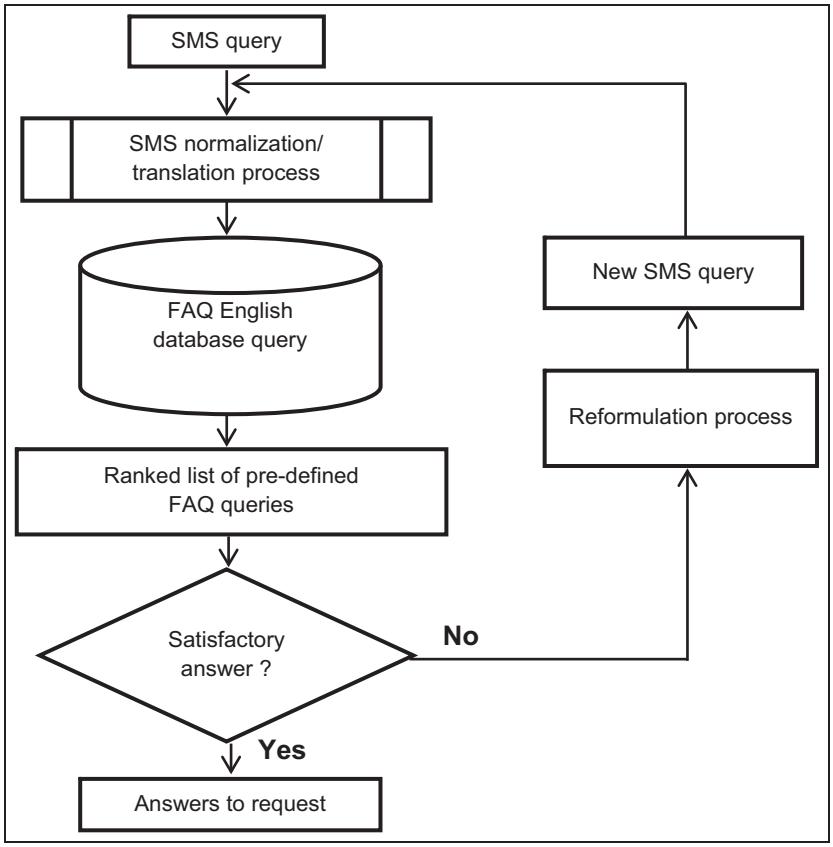

Figure 1. Flowchart of SMS-based retrieval system (SMS question locator – *SMSql*).

4. Research problem and methodology

This section discusses the research problem and the approach adopted to solving the problems.

4.1. Research problem

The research sets out to investigate ways of receiving an accurate response when an SMS text is used as a query to a FAQ database server in order to garner advice on a specific health domain. The research focused on two research questions which are:

(1) How should the design and development of an SMS-based information access system be achieved?
(2) How is the retrieval efficiency of the developed SMS-based information access system determined?

4.2. Research methodology

The experiment involved the use of FAQ database consisting of over 350 sampled questions on issues of HIV/AIDS – drug administration, prevention, control and support, counselling, food prescription, awareness, sex education, and education and training. The FAQ dataset comprised English words and HIV/AIDS terminologies. Out of these sampled questions, about 200 questions were extracted from Ipoletse call centres [50] and the remainder were gathered from over 15 related websites. Ipoletse data is a question resource of 205 most frequently asked questions about HIV/AIDS and antiretroviral therapy. The booklet [50] was prepared by the Ministry of Health, Botswana. Several other SMS-based retrieval process researches on the HIV/AIDS FAQ system make use of Ipoletse documents [16, 36, 46, 51]. The related websites have vast information on the HIV/AIDS epidemic in FAQ forms on aspects of drug administration, therapy, sex education, food and nutrition, physical exercise and treatment. The data was collected over a period of 20 months.

For the purpose of keyword extraction, 350 sampled questions used in this experiment were presented, individually, to about 140 students of the University of the Western Cape. One-hundred sample booklets were recovered from the students. These were used for analysis. The students were educated on the purpose of the research. The likely terms, idioms or words that can be recognized as the keywords per FAQ query sentence were identified by the students. The terms, idioms or words with the greatest frequency were selected as the keywords. This was achievable by statistically selecting *keywords* and *idioms* from the query corpus in the FAQ query-set gathered earlier. The keywords and idioms are combinations of words or phrases that give a reasonable meaning to each query. From the keyword phrases, idioms can be derived. For example, a query sentence 'when do you initiate antiretroviral therapy' has three statistically selected words – *initiate, antiretroviral* and *therapy* – as the idiom equivalents or keyword phrases. An *idiom* on the other hand is a collection of words with a specific semantic meaning as a group, which may not yield the same meaning when interpreted individually as words and not collectively as a phrase [52].

From the FAQ query set, 20 questions were statistically selected for our analysis. These questions were translated to SMS shorthand by the students of the University of the Western Cape. A set of 20 questions from 10 respondents yielded the 200 SMS-query formats used in our dataset, that is, each query had 10 respondents. A large collection of data was necessary in order to reduce the tendency towards bias in the SMS writing. Extraction of the best match question–answer pairs in the FAQ server was the ultimate goal. Keywords for each question were extracted based on the frequency level from more than 10 respondents. This meant that each question had a maximum of 10 SMS variants, that is, the number of respondents. In summary, there were 65 questions with similar keywords and idiom equivalents from the 350 questions originally gathered for this experiment.

At this stage it is important to note that stop words were less important parts of the keyword phrases and were discarded. Stop words are very common words that appear frequently in text and carry little or no semantic meaning in an expression [53]. Stop words affect the retrieval effectiveness because they have high frequency and tend to diminish the impact of frequency differences among less common words, affecting the weighting process [54]. It is therefore recommended that high-frequency word *n*-grams that occur in many words be eliminated before computing the similarity coefficient. Weighting the remaining *n*-grams using an inverse frequency coefficient, that is, assigning the highest values to the least frequently appearing *n*-grams, will ensure that matches between less frequent *n*-grams contribute more to word similarity than matches between frequent *n*-grams [55].

The retrieval efficiency results of the two algorithms – *term frequency–inverse document frequency* (*tf-idf*) and *SMSql* – are used as the basis for judging of the efficient algorithm. The scale of relevance judgement needed in calculating the retrieval efficiency was placed on a scale of 5, where excellent = 5; very good = 4; good = 3; moderate = 2; and poor = 1. The judgement was based on the first five FAQ sets of queries that emerged from the various ways in which SMS questions were sent into the search engine. This approach is similar to that of Mogadala *et al.* [38], where *cleaned* SMS texts

were used as a query to match the five best documents containing the FAQ question using the language model approach. It is important to map the position of the SMS query to the way the FAQ questions are presented in each of the algorithms being compared. A maximum of 5 points was allotted to an SMS enquiry that exactly produced the intention of the SMS *texter* in terms of the FAQ dataset. A value of 0 points was considered for the situation of out-of-domain, whereby the result of the FAQ query was completely different from the SMS enquiry. Some SMS queries were out-of-domain and did not have any corresponding FAQ answer [26, 38]. The next section presents the two algorithms used in the SMS-based information accessing techniques.

5. Proposed algorithm – SMS question locator (*SMSql*)

This section discusses the *SMSql* algorithm over the SMS FAQ search and retrieval system for mobile communication. The normalized idiom equivalents or keyword phrases extracted from the SMS query were matched with words present in our FAQ corpus. The algorithm considered similarity in words between the SMS query and the FAQ database, the length of the two sentences and the order in which the words were placed. The length of the query sentence was given priority. For easy identification each question (with its corresponding answers) had a unique code. Keyword isolation and identification led to further derivation of idioms. The interpretation of the wordings was individually done but considered collectively for FAQ query selection.

One of the methods adopted in arriving at a ranked list was assigning weights to the relevant terms. This showed the degree of importance of the terms (tokens) in the documents. Weight difference was needed for the following reasons: (1) to measure the degree of similarity between the FAQ terms and SMS query terms; (2) to determine the length and specificity of the query sentences; and (3) to determine the number of relevant FAQ documents (sentences), that is, the number of query sentence terms. A weight function/value of 1 was used to measure the similarity between the FAQ terms and the SMS query terms. A weight function/value of 2 was used to confirm the FAQ query sentence length. The keyword terms that were available in the FAQ sentence (and non-matching) were assigned 2. This was important if there was to be a tie in the weight function between FAQ terms and SMS query. The FAQ query sentence with lower sum of non-matching terms was considered as the chosen FAQ query sentence. Figure 2 provides a step-by-step description of the *SMSql* algorithm

SMSql processed the input sentence word-by-word from left to right. When the first SMS word (target word) was found, then the context window was built. This window was formed by the words placed just before and after the target word present in the FAQ database. The window size used in our system was 3, which included the target word and one word to its left and right, following the claim by Huang et al. [56] and Michelizzi [57] that words further away from the target word are less likely to be related than words close to the target word.

When a FAQ file was chosen as the query was being issued, the system iterated through the QA pairs in the file, comparing each question against the user's question and computing a score based on the *weight function*. We defined a *scoring function* for assigning a score to each statistically selected keyword phrase in the question corpus Q, where SMS token s_i had been normalized to English term t in the dictionary. Therefore, there was a similarity measure $\S$, between s_i and t such that $\S(s_i, t) > 0$ and this was denoted in the equation as $s_i \approx t$ in the equation. The *score function* measured how closely the question matched the SMS question string S.

Consider a query term q in every FAQ query sentence in the overall dataset Q as $q \in Q$, in the particular query sentence for each token SMS string s_i, the *scoring function* chose the term from q having the maximum weight. Then the weights n of the chosen terms were summed together, which gave the score:

$$\text{Score}(Q) = \sum_{i=1}^{n} \left[\underset{t:t \in Q \text{ and } s_i \sim t}{\max} w(s_i, t) \right] \quad (1)$$

The goal was to efficiently find the best matches to the query in the FAQ. The five selections with the highest scores were selected and were returned to the user. Each question from the FAQ file was matched against the user's question and then scored. Figure 3 illustrates the step-by-step description of the *tf-idf* algorithm.

Step 1	A weight function/value of 1 is assigned for equal matches of the two terms in the FAQ database and the English query term, otherwise it is set to 2 for other non-matching tokens.
Step 2	Sum the assigned values of matches in the FAQ query.
Step 3	Sum the assigned values of non-matching tokens in the FAQ query.
Step 4	Rank the weight function/value (in Step 2) in decreasing order.
Step 5	In case there is a tie in Step 2, select the FAQ query sentence with lowest sum non-matching tokens.
Step 6	Output the five best ranked query codes.

Figure 2. The SMSql algorithm.

Step 1	*Document pre-processing steps* Tokenization – a document is treated as a string, or bag of words, and then partitioned into a list of tokens. Frequently occurring or insignificant words, i.e. stop words, are eliminated. Stemming word - this step is the process of conflating tokens to their root form, e.g. *correct* for *correction*, *correcting*, *corrects*, *corrected*
Step 2	*Document representation* *n*-distinct words from the SMS and FAQ corpora are statistically selected. The collections are represented as the *n*-dimensional vector term space.
Step 3	*Computing term weights*. Get term frequency (*tf*). Find inverse document frequency (*idf*). Compute the *tf-idf* weighting.
Step 4	*Measure similarity between two documents* (SMS query and FAQ dataset) Calculate the *cosine similarity* by determining the cosine of the angle between two document vectors.

Figure 3. The *tf-idf* algorithm.

Using the *tf-idf* algorithm we were able to perform the ranking of the FAQ query for the set of SMS queries given by 10 SMS users over 20 questions. This was ranked and represented as relevance of the questions based on the SMS enquiries for this approach.

6. Performance evaluation

Average precision and *average recall* were the means of the *precision* and *recall* values obtained respectively from the set of top k (k was the size of FAQ query document) existing in FAQ datasets after each relevant FAQ query was retrieved, and this value was then averaged over information needs. That is, the set of relevant FAQ query documents for an information need $q_j \in Q$ is $\{d_1, \ldots, d_{m_j}\}$ and R_{jk} is the set of ranked retrieval results from the top result until the FAQ query document d_k is retrieved.

The best way to test the performance of different retrieval strategies is by using a simulation experiment. In this setting, a sample of queries is available and the documents that are relevant to each query have already been statistically identified. The performance of each automatic system can then be compared with a known standard of optimal performance. Systems are rated according to their ability to rank the relevant documents higher than the documents that are not relevant. While one can give a number of arguments about how and why this test setting does not reflect reality, no better methods for evaluating performance have been developed [58].

The efficiency of the retrieval mechanism was determined by the system retrieval and learning performance. The best retrieval strategy depends greatly on the length and specificity of the query because a complex data-driven retrieval strategy has little success with short queries and limited amounts of information [59]. Users of search engines have been accustomed to using short queries with keyword combinations owing to the restriction of the interface and inner mechanism of the search engine [59]. However, the detail that they provided might be vital to obtain good results for longer, more precisely defined queries where little vocabulary is shared by relevant documents, so that the system may be required to have some language understanding capability in order to discover relevant answer documents [60].

Therefore retrieval efficiency can be calculated through *precision* and *recall*. The learning performance involves performing the same set of experiments with a pre-determined number of iterations with the same dataset a particular number of times. To conduct the evaluation, the following steps were taken:

- We took a sample of 20 SMS coded FAQ query sentences. (*Mostly they were a set of queries that had greater representation from the data collected from the respondents, and were determined statistically.*)
- Each query was designed to retrieve the five best answers. The results were verified by experienced users using datasets applied at the beginning of our experiment and their corresponding answers.
- The retrieval efficiency was measured using precision and recall.
- The computational speed of the two algorithms was compared in order to determine the technique that is faster.

Precision is the relative number of correct constituents (FAQ query) retrieved from those retrieved as relevant. Hence the value must be as high as possible for good parsing. A constituent is considered to be correct if it matches a constituent in the Gold Standard (the structure representing the ideal analysis which the parsing results intended [61]).

$$\text{Precision} = \frac{\text{Number of relevant FAQ queries}}{\text{Number of retrieved FAQ queries}} \quad (2)$$

Recall is the relative number of correct constituents compared with the gold standard parse. It shows how many relevant answers were actually retrieved out of the possible answers – the higher the recall value is, the better the algorithm performance is.

The two metrics, *precision* and *recall*, which are inversely related, are computed using unordered list of FAQ query sets [62]. They are based on the user's relevance assessments following the retrieval process [60]. Therefore, the automatic handling of the various forms of user queries requires not only a large database of QA pairs but also the technology to match the user query to the FAQ documents in the database [40].

7. Results

The evaluations for this experiment were carried out in duplicate. They are presented as follows:

7.1. Comparison between tf-idf and SMSql algorithms in terms of average precision and average recall

The results of the experiment are given in Figures 4 and 5, where the average precision and average recall are plotted against a set of selected queries. If 10 SMS users send the same SMS query sentence to the search engine, the SMS writing of each user varies. The average precision and average recall were taken for the 10 SMS users (for each query sentence) based on the relevance judgement. Overall, the performance of SMS queries in the *SMSql* and *tf-idf* techniques appeared very difficult to confirm. Thus there was a need to perform a statistical test on the results and confirm the significance.

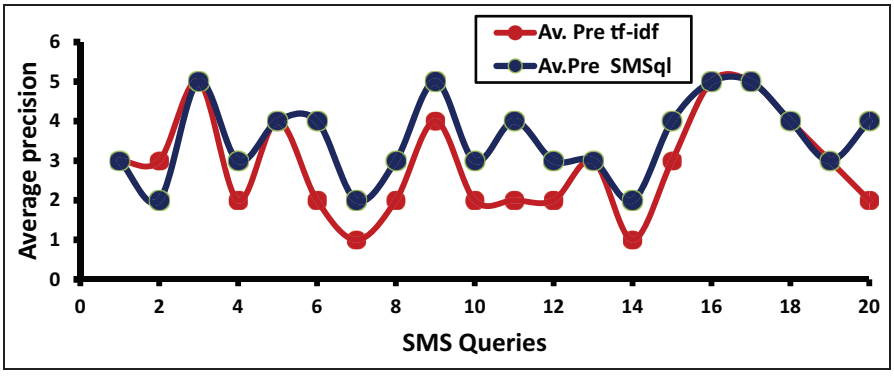

Figure 4. Average precisions for SMS queries in the *SMSql* and *tf-idf* algorithms.

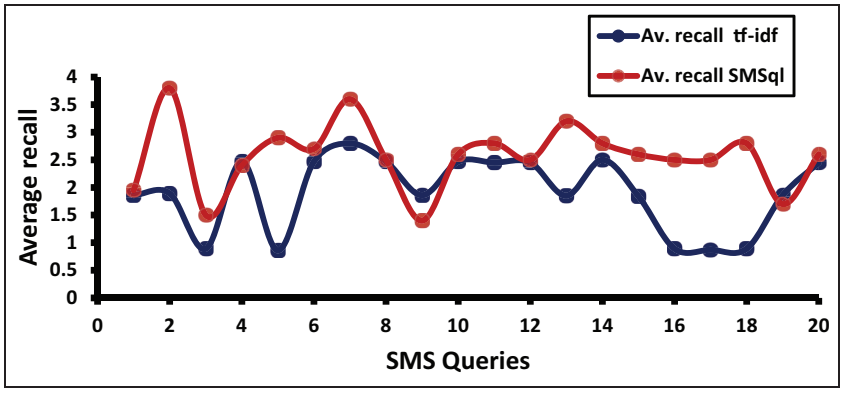

Figure 5. Average recalls for SMS queries in the *SMSql* and *tf-idf* algorithms.

The significance test was adopted to reject the null hypothesis, H_o, which means that there is no difference between the results of the two methods. This is done by comparison of the mean precision values across all the queries. The *t-test* was used to compare the mean scores for *same* group (of 10 users) and *same* condition or method on two different occasions, or when there are matched pairs [63]. A paired-samples *t-test* was conducted to evaluate the average precision for *SMSql* and *tf-idf* algorithms. There was a statistically significant difference in the performance of *SMSql* (mean = 3.55, standard deviation = 0.9987) and *tf-idf* (mean = 2.90, standard deviation = 1.2524); $t(19) = -3.577, p > 0.005$ (p-value = 0.002) at a confidence interval of 95%.

7.2. Comparison between tf-idf and SMSql algorithms in terms of average computational time

These metrics (average precision and recall) may not be sufficient to prove the algorithms. The system was further tested by employing a timing computation of the retrieval system. The two results were compared using the computational time. The results are shown in Table 1. The two algorithms were also compared in terms of computational speed. The computational speed was measured by estimating the time it took to retrieve the best five FAQ documents for each SMS query sentence. The total time used by the 10 users was then averaged and was taken as the average computational time for the SMS query iteration. The estimate was taken for the two algorithms and the average computational time was recorded for the methods. This is similar to the work described by Pudil et al. [64] using simple feature selection. The methods of Pudil et al. [64] show similar performance and differ only in computational efficiency. The objective of the comparison was identifying sub-optimal search methods. This can be achieved by considering computational time and efficiency.

In order to demonstrate clearly the effectiveness of each method, the selection of a feature set from the data showing high statistical dependencies provides a more discriminating test [64]. The execution time to generate results was compared for the two algorithms. The system of calculating the execution time can be constructed out of sequential programs but are typically built from concurrent programs called tasks [65].

The percentage of improvement between *SMSql* and *tf-idf* was 4.1%, that is, $(0.073 - 0.07)/0.073*100\% = 4.1\%$. The results proved that the *SMSql* is 4.1% better than the *tf-idf* approach using the computational time as the metric value. From Table 1, the average time taken (t_n) for an SMS query number (Q_n) for each SMS request by the user was taken for each of the algorithms, and the results presented. The score of each query sentence was calculated sequentially and then ordered to generate the result. The average time for each iteration of the SMS queries, SMS Q_1–Q_{20}, for each algorithm was taken. The results show the time spent in generating responses to requests made in this experiment.

Table 1. Time computation for the retrieval process of the SMS queries.

SMS query no. (Q_n)	Average computational time for 10 iterations per SMS query	
	tf-idf (s)	*SMSql* (s)
1	0.085	0.097
2	0.077	0.087
3	0.086	0.076
4	0.074	0.072
5	0.085	0.085
6	0.037	0.037
7	0.069	0.069
8	0.067	0.067
9	0.077	0.072
10	0.068	0.068
11	0.074	0.074
12	0.088	0.078
13	0.067	0.057
14	0.075	0.075
15	0.082	0.062
16	0.067	0.057
17	0.078	0.078
18	0.068	0.062
19	0.059	0.059
20	0.074	0.064
Total	1.457	1.396
Average	0.073	0.070

8. Conclusion

It is imperative to link information seekers to information sources by matching the query with the description of the content that is associated with the indexed information segments in the database. This paper investigated a situation where SMS text is used to make an enquiry from the search engine on health-related matters. Information seekers are not patient enough to source results beyond the preliminary download pages, and unfortunately, the *noisy* form in which SMS appears is not sufficient to provide accurate results until it is normalized. This creates a great challenge to the information seeker whenever SMS is used to search for information. An algorithm was developed to use the normalized form of SMS text for information search. The developed algorithm considered the similarity in words between the SMS query and the FAQ database, the length of the two sentences as well as the order in which the words are placed. The retrieval efficiency of the developed algorithm was compared with the existing algorithm in an SMS-based FAQ system. The query-term similarity between the SMS and FAQ was used to present the five best ranked relevant results. Three parameters – average precision, recall and computational time – were used for the basis of proving that the developed method was better when the retrieval efficacy was considered. Statistically, there was significant difference in the performance of the two techniques and also the computational speed of the developed algorithm proved to be better by 4%. The developed algorithm has been proved to produce fast results for the normalized SMS-query.

Acknowledgements

Our appreciation goes to Professor Isabella M. Venter for her contribution in the early stages of producing the manuscript.

Funding

The authors would like to acknowledge the Senate Research Committee of the University of the Western Cape, Bellville, South Africa for funding.

References

[1] Pass G, Chowdhury A and Torgeson C. A picture of search. In: *First international conference on scalable information systems*, 2006.
[2] Rose DE and Levinson D. Understanding user goals in web search. In: *Proceedings of the 13th international conference on the World Wide Web*. New York: ACM, 2004, pp. 13–19.
[3] Fairon C and Paumier S. A translated corpus of 30,000 French SMS. In: *Proceedings of the fifth international conference on language resources and evaluation (LREC 2006)*, Sweden, 2006, pp. 351–354.
[4] Chen J, Subramanian L and Brewer E. SMS-based mobile web search for low-end phones. In: *16th Annual international conference on mobile computing and networking*. New York: ACM, 2010, pp. 125–135.
[5] Chen J, Linn B and Subramanian L. SMS-based contextual web search. In: *Proceedings of the 1st ACM workshop on networking, systems, and applications for mobile handhelds*. New York: ACM, 2009, pp. 19–24.
[6] Cormick G, Kim NA, Rodgers A, Gibbons L, Buekens PM, Belizán JM et al. Interest of pregnant women in the use of SMS (short message service) text messages for the improvement of perinatal and postnatal care. *Reproductive Health* 2012; 9(1): 1–7.
[7] Mendez J and Maher J. Evidence Supporting the use of text messaging for partner services. *Sexually Transmitted Diseases* 2012; 39(3): 238–239.
[8] Lester RT, Ritvo P, Mills EJ, Kariri A, Karanja S, Chung MH et al. Effects of a mobile phone short message service on antiretroviral treatment adherence in Kenya (WelTel Kenya1): A randomised trial. *The Lancet* 2010; 376(9755): 1838–1845.
[9] Armstrong K, Liu F, Seymour A, Mazhani L, Littman-Quinn R, Fontelo P et al. Evaluation of txt2MEDLINE and development of short messaging service–optimized, clinical practice guidelines in Botswana. *Telemedicine and e-Health* 2012; 18(1): 14–17.
[10] Meingast M, Roosta T and Sastry S. Security and privacy issues with health care information technology. In: *Proceedings of the 28th IEEE EMBS annual international conference*, New York, 30 August to 3 September 2006, pp. 5453–5458.
[11] Kaplan WA. Can the ubiquitous power of mobile phones be used to improve health outcomes in developing countries. *Global Health* 2006; 2(9).
[12] Déglise C, Suggs LS and Odermatt P. SMS for disease control in developing countries: A systematic review of mobile health applications. *Journal of Telemedicine and Telecare* 2012; 18(5): 273–281.
[13] Zaman M. Integrating MCT with RFID: A case study. *Journal of Global Research in Computer Science* 2013; 4(3): 80–82.
[14] West D. How mobile devices are transforming healthcare. *Issues in Technology Innovation*, 2012.
[15] Blynn E. *Piloting mHealth: A research scan*. Cambridge, MA: Knowledge Exchange Management Sciences for Health, 2009.
[16] Masizana-Katongo AN, Anderson G, Mpoeleng D, Taukobong T, Mosweunyane G, Eyitayo OT et al. An SMS-based healthcare information storage and retrieval system. In: *Proceedings of the IASTED African conference health informatics (AfricaHI2010)*, Gaborone, Botswana, 2010.

[17] Lin D. *An information-theoretic definition of similarity*. ICML, 1998, pp. 296–304.
[18] Kondrak G, Marcu D and Knight K. Cognates can improve statistical translation models. In: *Human language technology conference of the North American chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*, Edmonton, 2003, pp. 46–48.
[19] Danish Contractor, Kothari G, Faruquie TA, Subramaniam LV and Negi S. Handling noisy queries in cross language FAQ retrieval. In: *Proceedings of the 2010 conference on empirical methods in natural language processing*. Cambridge, MA: MIT/Association for Computational Linguistics, 2010, pp. 87–96.
[20] Govind K, Negi S, Faruquie TA, Chakaravarthy VT and Subramaniam LV. SMS based interface for FAQ retrieval. In: *Proceedings of the 47th annual meeting of the ACL and the 4th IJCNLP of the AFNLP*, Suntec, Singapore, 2009, pp. 852–860.
[21] Chabossou A, Stork C, Stork M and Zahonogo P. Mobile telephony access & usage in Africa. In: *Proceedings of the international conference on information and communication technologies and development (ICTD)*, Doha. New York: IEEE, 2009.
[22] Hellström J. The innovative use of mobile applications in East Africa. *Sida Review* 12, 2010.
[23] Tomitsch M, Sturm F, Konzett M, Bolin A, Wagner I and Grechenig T. Stories from the field: Mobile phone usage and its impact on people's lives in East Africa. In: *Proceedings of the international conference on information and communication technologies and development (ICTD'10)*, 2010.
[24] Edosomwan J and Edosomwan TO. Comparative analysis of some search engines. *South African Journal of Science* 2010; 106(11/12): 1–4.
[25] Ahmed F, De Luca EW and Nürnberger A. Revised n-gram based automatic spelling correction tool to improve retrieval effectiveness. *Polibits* 2009; 40: 39–48.
[26] Hogan D, Leveling J, Wang H, Ferguson P and Gurrin C. DCU@ FIRE 2011: SMS-based FAQ retrieval. In: *3rd Workshop of the Forum for Information Retrieval Evaluation, FIRE*, 2011, pp. 2–4.
[27] Beaufort R, Roekhaut S, Cougnon L-A and Fairon C. A hybrid rule/model-based finite-state framework for normalizing SMS messages. In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguists*, Uppsala, Sweden, 2010, pp. 770–779.
[28] Pakray P, Pal S, Poria S, Bandyopadhyay S and Gelbukh A. SMSFR: SMS-based FAQ retrieval system. *Advances in Computational Intelligence*. Berlin: Springer, 2013, pp. 36–45.
[29] Leveling J. DCU@ FIRE 2012: Monolingual and crosslingual SMS-based FAQ retrieval. In: *FIRE 2012, fourth workshop of the forum for information retrieval evaluation*, Kolkata, India, 4–6 December 2012, pp. 37–42.
[30] Budzik J and Hammond KJ User interactions with everyday applications as context for just-in-time information access. In: *Proceedings of the 5th international conference on Intelligent user interfaces*. New York: ACM, 2000, pp. 44–51.
[31] Hajek P and Valdés JJ. An analysis of MYCIN-like expert systems. *Mathware and Soft Computing* 2008; 1(1): 45–68.
[32] Witten IH and Madams PH. The Telephone Enquiry Service: A man–machine system using synthetic speech. *International Journal of Man–Machine Studies* 1977; 9(4): 449–464.
[33] Evans RE. The impact of voice characteristics on user response in an interactive voice response system 2009, Masters dissertation, Rice University, Houston, TX.
[34] Hammond K, Burke R, Martin C and Lytinen S. FAQ finder: A case-based approach to knowledge navigation. In: *Proceedings of 11th conference on artificial intelligence for applications*, 1995, pp. 80–86.
[35] Burke RD, Hammond KJ, Kulyukin V, Lytinen SL, Tomuro N and Schoenberg S. Question answering from frequently asked question files: Experiences with the faq finder system. *AI Magazine* 1997; 18(2): 57.
[36] Thuma E, Rogers S and Ounis I. Evaluating bad query abandonment in an iterative SMS-based FAQ retrieval system. In: *Proceedings of the 10th conference on open research areas in information retrieval*, 2013, pp. 117–120.
[37] Burke RD, Hammond KJ, Kulyukin V, Lytinen SL, Tomuro N and Schoenberg S. Question answering from frequently asked question files: Experiences with the FAQ finder system. *AI Magazine* 1997; 18(2): 57.
[38] Mogadala A, Rambhoopal K and Varma V. Language modeling approach to retrieval for SMS and FAQ matching, 2012.
[39] Prochasson E, Viard-Gaudin C and Morin E. Language models for handwritten short message services. In: *Ninth International Conference on Document Analysis and Recognition, ICDAR*. New York: IEEE, 2007, Vol. 1, pp. 83–87.
[40] Langer A, Banga R, Mittal A and Subramaniam LV. Variant search and syntactic tree similarity based approach to retrieve matching questions for SMS queries. In: *AND'* 10, 2010.
[41] Jain M. *N-Gram driven SMS based FAQ retrieval system*. Delhi: Delhi College of Engineering Delhi University. 2012.
[42] Joshi A. Improving accuracy of SMS based FAQ retrieval. *International Journal of Emerging Technologies in Computational and Applied Sciences* 2012; 362–366.
[43] Acharyya S, Negi S, Subramaniam L and Roy S. Unsupervised learning of multilingual short message service (sms) dialect from noisy examples. In: *Proceedings of the second workshop on analytics for noisy unstructured text data*. New York: ACM, 2008, pp. 67–74.
[44] Cook P and Stevenson S. An unsupervised model for text message normalization. In: *Proceedings of the workshop on computational approaches to linguistic creativity*. ACL: Boulder, CO, 2009, pp. 71–78.
[45] Anderson G, Ayalew Y, Mokotedi PA, Motlogelwa NP, Mpoeleng D and Thuma E. Healthcare FAQ information retrieval using a commercial database management system. In: *Proceedings of the 2nd IASTED Africa conference on modelling and simulation (AfricaMS 2008)*, Gaborone, Botswana, 2010.

[46] Masizana-Katongo A, Anderson G and Mpoeleng D. Healthcare FAQ information retrieval using SMS language. In: *Prato CIRN-DIAC community informatics conference*, 2010, Refereed Stream.
[47] Chuklin A and Serdyukov P. Good abandonments in factoid queries. In: *Proceedings of the 21st international conference companion on World Wide Web*. New York: ACM, 2012, pp. 483–484.
[48] Koumpouri A and Simaki V. Queries without clicks: evaluating retrieval effectiveness based on user feedback. In: *Proceedings of the 35th international ACM SIGIR conference on research and development in information retrieval*. New York: ACM, 2012, pp. 1133–1134.
[49] Li J, Huffman S and Tokuda A. Good abandonment in mobile and PC internet search. In: *Proceedings of the 32nd international ACM SIGIR conference on research and development in information retrieval*. New York: ACM, 2009, pp. 43–50.
[50] Ipoletse. *Ipoletse training manual for call centre operators for the National Call Centre on HIV and AIDS*, Gaborone, Botswana. 2002.
[51] Anderson G, Asare SD, Ayalew Y, Garg D, Gopolang B, Masizana-Katongo A et al. Towards a bilingual SMS parser for HIV/AIDS information retrieval in Botswana. In: *Proceedings of the second IEEE/ACM international conference of information and communication technologies and development (ICTD)*, Bangalore, India, 2007, pp. 329–333.
[52] *Oxford advanced learner's dictionary of current English*, 7th edn. Oxford: Oxford University Press, 2006.
[53] Dragut E, Fang F, Sistla P, Yu C and Meng W. Stop word and related problems in web interface integration. In: *Proceedings of the VLDB Endowment* 2009; 2(1): 349–360.
[54] Abu El-Khair I. Effects of stop words elimination for arabic information retrieval: A comparative study. *International Journal of Computing and Information Sciences* 2006; 4(3): 119–133.
[55] Robertson AM and Willett P. Applications of n-grams in textual information systems. *Journal of Documentation* 1998; 54(1): 48–67.
[56] Huang LB, Balakrishnan V and Raj RG. Improving the relevancy of document search using the mult-term adjacency keyword-order model. *Malaysian Journal of Computer Science* 2012; 25: 1.
[57] Michelizzi J. Semantic relatedness applied to all words sense disambiguation. *University of Minnesota*, 2005.
[58] Hull DA. Information retrieval using statistical classification. *Stanford University*, 1994.
[59] Dayong W, Yu Z, Shiqi Z and Ting L. Identification of web query intent based on query text and web knowledge. In: *First international conference on pervasive computing, signal processing and applications*, Harbin, China, 2010, pp. 128–131.
[60] Maleki-Dizaji S. Evolutionary learning multi-agent based infromation retrieval systems. PhD Thesis, Sheffield Hallam University, 2003.
[61] Masizana-Katongo A and Ama-Njoku T. Example-based parsing solution for a HIV and AIDS FAQ system. *International Journal of Research and Reviews in Wireless Communications (IJRRWC)* 2011; 1(3): 59–65.
[62] Buckland M and Gey F. The relationship between recall and precision. *JASIS* 1994; 45(1): 12–19.
[63] Pallant J. *SPSS survival manual: A step by step guide to data analysis using SPSS*. Buckingham: Open University Press, 2010.
[64] Pudil P, Novovičová J and Kittler J. Floating search methods in feature selection. *Pattern Recognition Letters* 1994; 15(11): 1119–1125.
[65] Puschner P and Koza C. Calculating the maximum execution time of real-time programs. *Real-Time Systems* 1989; 1(2): 159–176.