Journal of the Nigerian Association of Mathematical Physics
Volume 53 **(November *2019 Issue*), pp153 – 168**

SMS-BASED HEALTHCARE FAQ INFORMATION RETRIEVAL SYSTEM

Ademola Olusola Adesina

**Department of Mathematical Sciences, (Computer Science Unit),
Olabisi Onabanjo University, Ago Iwoye, Ogun State, Nigeria**

Abstract

The goal of efficient information retrieval in a question and answer database is to present the best answer to a user's request. If the information need is not met by the user there is a probability of search abandonment or query reformulation. This action may have come up as a result of the use of non-standard words created by the wrong spelling, acronym, abbreviation, and contractions that are un/deliberately used for the search. When this happens, relevant results are far presented for the information needs of the user hence leads to user's dissatisfaction. Short message service (SMS) is full of such features, hence motivates this research. The concern of this paper is to initiate an SMS-based query into a frequently asked question (FAQ) search engine in a healthcare domain. In the early stage of the experiment, the SMS word underwent the normalization process to become a normalized SMS word. The normalized SMS word is then used to query the search engine. An information retrieval algorithm was developed that works on the similarity between the normalized SMS terms and FAQ collections. The research questions consider the system design of a robust SMS-based FAQ architecture, the measurement of the retrieval qualities of the algorithm, and the speed of retrieval measured in terms of computational speed. The evaluation was carried out based on the three information retrieval metrics – precision, recall and retrieval time. The results are compared with the two existing retrieval algorithms.

***Keywords*:** Information retrieval (IR), Short message service (SMS), Text message, Frequently asked question (FAQ), HIV/AIDS, mobile health (mHealth), Question and answer (Q&A) system

1. Introduction

The trend and development of mobile communications have reflected in increasing utilization of cell phones and the use of short message service (SMS), also referred to as text message, in the healthcare services[1-3]. The text message capabilities in mobile phones can be used to provide adequate medical information on the awareness, prevention and treatment of some dreadful diseases. SMS common activity in *health matters* is noticeable in the area of appointment reminders, medication taking, telemedicine, accessing patient records, communication of laboratory test results, measuring treatment compliance, raising health awareness, monitoring patient's illness, and physician decision support [4, 5] and other virtual health assistants. Text message provides the missing link between a hospital and its field worker, patients, peer groups, or community health workers wherever they may be [6]. Patients make use of the SMS tool to keep in touch with their family and friends during hospitalization [7].SMS language 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.SMS is inexpensive in terms of cost and is applied quietly without disturbing other patients when they are not occupied with any medical procedures [7]. It offers an alternative or supplementary means of providing and receiving social support for patients in hospitals[7]. The role of SMS in mobile health services cannot be underestimated, for instance, it is used in South Africa to remind tuberculosis

Corresponding Author**:** Ademola O.A., Email: inadesina@gmail.com, Tel: +2348035359488

patients to administer their medication. *Rifafol*, the medication for tuberculosis, is expected to be taken daily and on a consistent basis, otherwise, it is not effective. SMS texts which are written in English and local languages—Afrikaans and Xhosa—are sent at a pre-determined time daily to patients. This is done for a period of six months until the treatment is completed [5].

As important as SMS is to health-related matters; it is difficult to use its full capabilities in mobile information access because of its features. SMS feature as created a unique pattern of writing, an invention of abbreviations, and the use of non-standard orthographic forms [8]. Its writing poses a great challenge to the transformation into the formal writing suitable for information processing like stemming, segmentation, regression, classification, clustering, and stops word identification. SMS language is non-aligned to the pattern of the traditional natural language because of the short form of writing, and the creativity of the *texters*. The SMS-based information access system involves the use of SMS as a source of input query sentence to the search engine. This retrieval technique may not be feasible unless a pre-processing stage called *SMS normalization* takes place. SMS normalization refers to the task of converting SMS text that could be *noisy* into its intended *non-noisy* form[9-11]. SMS normalization is beyond the scope of this paper.

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. Thuma et al. [12] in an iterative retrieval approach, suggested that several turns of query sentence may be needed so as to meet user's satisfaction in an SMS-based frequently asked question (FAQ) retrieval system. Sometime this leads to query abandonment, especially when the desire is not met in the early trials of the request. The iterative or reformulation process of the query may take longer time even discourage the user that may lead to abandonment of the enquiry. Users are known to surf through the first few web pages[13, 14].An SMS-based question answer (QA) retrieval system accesses information in the form of questions and answers with the use of SMS services on the mobile phone platform. QA systems may appear in four guises—(1) *natural language processing*– this is a situation whereby users send a query in a natural language for enquiries on phones or computer machine, and the answers are returned in a natural language (e.g. *www.google.com*), (2) *human intervention*– messages are sent in form of natural language to a particular agent. Normally, the agent who is an expert gives the answer to the request. There is areal-time answer to the question and the conversation can become interactive (e.g. *www.chacha.com*),(3) *information retrieval method*– the corpus will be searched for a possible answer to the request, the answer may be delivered after the enquirer has responded to the request from the machine, for instance, to press or type specific code to retrieve information, e.g. an enquiry from the mobile phone operators, and (4) *frequently asked question retrieval*– there is a ready-made answer to every enquiry that may be requested from the user. FAQ are users' questions and expert answers about specific domains [15] (e.g. *www.aids.gov/hiv-aids-basics*).FAQ query may be In-Domain (ID) or Out-of-Domain(OOD), ID query will have a corresponding answer to every query in the FAQ document unlike OOD [16].

For the remainder of this paper, the FAQ document will be referred to as the question-answer pair while the sample set of the normalized SMS query will be referred to as the FAQ query document collection. The FAQ query document collection is sent to interface with the set of FAQ documents. Each of the normalized SMS queries in the FAQ document collection has a corresponding query answer to the query in a one-to-one mapping. For example, a FAQ document may be structured as a query: "*what are Anti-retroviral drugs?"*answer: *"these are medications used to treat HIV build-up …"*
The emphasis laid on this paper is on the *FAQ retrieval system* where enquiries are made using the peculiar characteristics exhibited by text message. The research work is designed to give a set of FAQs for a query written in SMS language. The FAQ may be (1) *Mono-lingual FAQ retrieval*: the FAQ and SMS datasets are the same language; the challenge is to get the best matching between the two datasets. (2) *Cross-lingual FAQ retrieval*: the FAQ and SMS datasets are not the same language; the challenge is to get the best matching between two dissimilar datasets (3) *Multi-lingual FAQ retrieval*: the FAQ and SMS datasets are many languages; the challenge is to get the best matching between various languages or datasets. The database is searched for the SMS enquiry and an appropriately matched answer is returned. There is a need to have an overlap between the words in the two sentences. Monolingual SMS-based FAQ retrieval system is treated as the question answering system and the algorithm presented is on English.
The rest of 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 general architecture in building the SMS-query system is presented in Section 3. Section 4 discusses the research problems and data collection methods in developing the SMS-based FAQ system. Section 5 presents the keyword extraction algorithms and FAQ document query code identification. The research methodology is described in Section 6. The proposed SMS-based information search and retrieve algorithms and existing algorithms are described in Section 7. The results of the experiment are presented in Section 8. Finally, the paper is concluded in Section 9.

Related work

Significant work has been done on SMS-based FAQ systems [12, 17-19]. Different attempts with distinct contexts have been made to normalize text messages, like noisy channel modeling[20], phrase-based [21], character string [22], rule-based [19] etc. so as to get the SMS text available for other natural processing activities. Hogan et al. [19] identified SMS-based FAQ retrieval systems as having three steps (1) SMS normalization, (2) retrieval of ranked results, and (3) identifying out-of-domain query results. In order to normalize the SMS FAQ queries, a set of transformation rules were created and the corpora were manually annotated. The rules were never published. 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. Hogan et al. [19] examined each SMS tokenif it remains unchanged and then the corrected token is substituted. A set of candidate lists are generated. The best candidate in the context is selected as the correction. There are 3 methods for selecting the best candidate: (1) manually annotated data was used as a correction rule, to get the best translation for the SMS tokens. The frequency of use of the correction rules becomes a criterion for calculating the normalized weights of the replacement of SMS token in the corpus. (2) Candidate corrections were created by consonant skeletons. The mapping between the consonant skeletons and the words produces additional correction candidates for the query words. (3) Candidates are generated when all words in the corpus are compared with the prefix of the question words, to confirm if there is truncation. The three methods produce replacement candidate lists, which are merged by adding their weightings from their term frequency. The token scores are calculated using the maximum product of that weight and the *n*-gram score of the corrected token. The disadvantage of this model is that it uses a manual annotation of the dataset, which 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 is 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. All the tokens are processed.

SMSFind is an SMS-based information retrieval model that focuses on the problem of "appropriate information extraction" [23, 24]. It uses the conventional search engine in the back-end to provide an appropriate answer for the FAQ query document collection. SMSFind uses the normalised SMS queries; typically, the arrangement contains a term or a collection of consecutive terms in a query that provides a *hint* as to what the user is looking for. These FAQ query document collection query terms or a collection of consecutive terms are provided as the *hint* to facilitate the matching process of the question-answer system. The*hint*, provided by the user or automatically generated from the document, is used to address the information extraction problem. SMSFind uses this *hint* to address the information extraction problem as follows: Given the top search responses to a query from a search engine, SMSFind extracts snippets of text from within the neighbourhood of the *hint* in each response page. SMSFind scores snippets and ranks them across a variety of metrics. The *hint* extracted is used to determine the answer to the request. It is scored based on a top-*n* listfor each page and it is ranked altogether. The highest score is released as an answer to the request [24]. The use of *hints* in the algorithm is considered a supervised learning approach [25, 26] and it adds costs to generate and store. The research never considered the contextual information of the searches as it is limited to a single term (*hint*), which is not enough for the contextual information. The searching is limited to the constituent of the *hint*. The *hint* is domain specific information use to obtain results i.e. it is user-defined.

Vilariño et al.[27]recent work are based on the probability model of an SMS based FAQ retrieval system. Monolingual, cross lingual and multilingual approaches were implemented on the dataset from three sources(English, Hindi and Malayalam languages). SMS normalization was carried out initially by substituting each query term with the closest translation offered by a bilingual statistical dictionary. The dictionary was used to calculate the most frequent calculated term from a training corpus of the SMS query term that is associated with FAQ terms. The *Gizza++* tool was used to calculate the most frequent term through the use of the IBM-4 model, by using a training corpus composed of a set of aligned phrases (i.e. one SMS term to its corresponding FAQ term). IBM-4 model works on the relative re-ordering of previously translated words (*cepts*) [27-29]. The similarity among the SMS terms and each of the FAQ questions was calculated using the *Jaccard* similarity coefficient. *Jaccard* coefficient measures similarity value, *N*, between SMS and FAQ sets by calculating the size of the intersection divided by the size of the union of the sample sets[30, 31]. All values of FAQs above *N* was returned as the answer set of the FAQ. There are two shortcomings of this method, (1) the contextual information of the SMS query and FAQs are better measured by considering a phrase-basedapproach than being word and (2) the approach of measurement did not take into account the frequency of the terms among the documents that are compared.

SMSFR is a recent SMS-based searching technique developed by Pakray et al.[32]. It has a multi-lingual (English, Hindi, and Malayalam) feature with multi domain FAQ datasets similar to Vilariño et al.[27]. Bing spellchecker, a free source dictionary was used for the SMS normalization process. It involves the *unigram*matching, *bigram* matching, and *1-skip bigram* matching modules done on the SMS and FAQs dataset. The research has the goal of getting the best FAQ for the SMS query. In the monolingual technique, the rule-based system for the ranking of the candidate FAQ terms is applied. The system has four modules (pre-processing, unigram, bigram, and 1-skips bigram matching modules) for the normalization processes. Bing spellchecker module processes the SMS and FAQ dataset to search for the matching of the new word. The similarity in the word of the SMS and FAQ confirms the search. But if there is no match, *WordNet 3.0* is searched for hyponyms, synonyms, etc. This is an extra cost on FAQ dataset as it is assumed to be error free. *WordNet* is a lexical database for the English language that groups English words into sets of synonyms called synsets [33]. The bigram matching compares the match between the two statements by considering the bigram occurrences of their words. The two consecutive words in the two datasets are compared. If there is a match, the next consecutive bigram is searched, otherwise, the *WordNet* is searched for the bigram sequences of the SMS and FAQ. 1-skip and inverse bigram matching consider a sequence bigram with one gap between two words. For every similarity of the two words (SMS and FAQ) in the list of SMS (S') that is found on the inverse order of FAQs list (F'), a set of semantic rules is applied to confirm because the pairs are not rejected, however, the complete set of the rules are not given. In general, the output of the top five scores are used for the single SMSquery processes. The use of Bing speller can be considered to be restricted to only words in the dictionary, if it is not in the database the right answers are not provided even though it is economical because Bing speller is free software.

Healthcare FAQ information retrieval systems using SMS in the form of a Question and Answer (Q&A) System were recently proposed by Anderson et al. [34] and Masizana-Katongo et al. [35]. SMS users submit queries to the portal through a mobile phone interface. A parsing technique was proposed asa retrieval mechanism in matching the relevant answers [18].The parser extracts and processes keywords from the SMS input text. This leads to the matching of the SMS keywords to a relevance FAQ dataset. 20 HIV/AIDS questions written in English were written in SMS format. Frequently occurring SMS terms were extracted from each question. Every question can now be evaluated in its merit from the combination of the frequently occurring phrases and/or words within the phrases. This can be achieved through statistical analysis. The SMS input format in form of grammar is then parsed through the automatic parser generator or compiler. A parser generator reads a grammar specification and converts it to program that recognize matches to the grammar. A method is generated (in the code) that corresponds to each production in the grammar. The technique involves the translation of the grammar provided in *Backus-Naur Form*(BNF) format into pre-processed parsed tree building blocks that can be easily implemented in Javacode. The system is evaluated using recall, precision, and rejection. Their procedure did not consider the ranking of the SMSquery in presenting the answer.

Kothari et al. [36] designed an automatic FAQ-based question answering system. The method involves promoting SMS query similarity to FAQ-questions. This is done through a combinatorial search approach. The search space consists of combinations of all possible dictionary variations of tokens in the noisy query. The combinatorial search system models an SMS query as a syntactic tree matching so as to improve the ranking scheme after candidate words have been identified. Initial processing of noise removal was introduced so as to improve information retrieval efficiency. The model involves the use of a dictionary and maps the SMS query to the questions in the corpus. The noise removal step is, however, computationally expensive [37]. However, the system developed by Kothari et al.'s[36] does not involve training SMS data on text normalization. It has the advantage of handling semantic variations in question formulation but the method fails to discuss the choice of homophonic words in the context of automatic speech recognition. Kothari et al. [36]depend 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 [38] and Joshi [39].

Anderson et al. [34] and Masizana-Katongo et al. [35] researches share similarities with ours in the area of application, i.e., health related matters. But the two research groups use SMS parsing techniques to query the search engine after the SMS token has been disambiguated using a context-free grammar. In our approach, an SMS term is taken as a query while the FAQ is considered as a document for the SMS-based retrieval. Their research was applied to a multilingual scenario whereas we considered English only. The general architecture overview for the proposed SMS-based information retrieval system is discussed in the next section.

General architecture overview

Figure1 shows how the SMS query is presented to the web search engine. The mobile phone user sends an SMS message to the server which is processed as a search query within the question answer (QA) database. The SMS query sentence, in a normalized SMS form, is made to interface with the FAQ-SMS database. The web server contains the FAQ-SMS database, predefined questions, and corresponding answers to the queries. The set of predefined questions are English query versions of the various forms in which SMS queries are assumed to be used by mobile phone users. There is also the corresponding set of answers to the predefined questions. The set of FAQ collections relevant to the SMS request are extracted through similarity computation, matching processing and inferences in order to meet the need of the user before a set of retrieved documents can be presented [40, 41]. The set of retrieved documents (answers) may sometimes be relevant or irrelevant to the user's needs. In this case, the query may need to be reformulated through the reformulation process[12, 42]. Query reformulation process is a key task towards next generation web search engines as it makes to predict user intent, providing the needed assistance at the right time, and thus helping users locate information more effectively and improving their web-search experience [43]. Every time a new set of query words is applied, with the same concept (semantics) in mind, a new crop of documents (answers) are retrieved and presented.

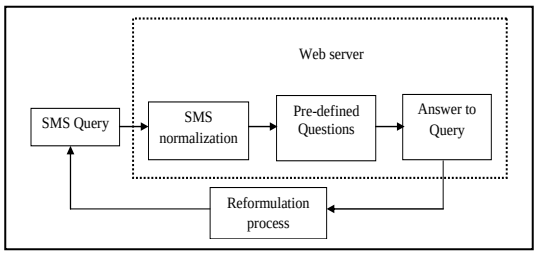

Figure 1. The system architecture of an SMS-Query and reformulation process

The methodology adapted to access information using cell-phones is done with the proposed *SMSql* algorithm on a web server which automates the answer retrieval task as illustrated in Figure 1. The retrieval process entails providing the five topmost relevant answers for a user enquiry. The baseline of five topmost relevant answers has been used for the evaluation in QA system in other experiments on the information retrieval system [44, 45].

4. Research problem and data collection

This section discusses the research problem and data collection methods.

4.1 Research problem

The research sets out to investigate two important issues: (1) How should an accurate response be received when an SMS text is used as a query to a FAQ database server in order to garner advice on a specific health domain and (2) How should the retrieval efficiency of the developed algorithm be measured compared to the existing algorithms.

4.2 Data collection

There are various ways that can be used to collect datasets for an experiment. For instance, the experiment performed by Jansen et al. [46] used log files where 74 terms were found to occur more frequently in their sample space of an average term of 100 using the Excite search engine. A collection of 1400 documents, from a United Nations database of 1988, were used in an experiment titled *using tf-idf to determine word relevance in document queries*. From the document, 86 queries were extracted to perform the experiment on information retrieval [47]. Burke et al. [48]used a total of 241 test questions from a corpus drawn from system log files. A widely read and popular news medium, *The Times of India* blog, was used as the source of data. The blog has several datasets on topics like politics, sports, entertainment, cuisine, social evils [49]. In our experiment the developed algorithms were tested and validated using the corpus prepared and collected in three different ways:

1. 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. Out of these sampled questions, about 200 questions were extracted from Ipoletse call centres [50] and the remaining were fetched from related websites. Ipoletse data is a question resource of 205 most frequently asked questions about HIV/AIDS and antiretroviral therapy. The booklet was prepared by the Ministry of Health in Botswana. Several other SMS-based retrieval process kinds of research on HIV/AIDS FAQ system make use of Ipoletse Q&A documents, like Anderson et al.[18], Masizana-Katongo et al. [51],and Thuma et al. [12]. The forty-two page booklet is titled *Training Manual for Call Centre*

Operators for the National Call Centre on HIV/AIDS with nine chapters covering different issues on HIV/AIDS. It is arranged in the form of question and answer, for example:

Question: What are my responsibilities as an HIV infected person?
Answer: *There is no clear-cut definition when it comes to HIV and sexual relationships. Some would say that the main responsibility is to always tell your partner that you have HIV, use condoms or other protection, and if possible, avoid sex without one. In reality, the responsibility is shared between both infected and uninfected people to strive toward avoiding transmission, whilst continuing to have as full and normal a life as possible.*

Question: What is counselling?
Answer: *Counselling is a helping relationship or a dialogue that intends to help one to make informed decisions. It is a safe and confidential place where you can talk about your life and the ways in which it can be affected by HIV.*

2. The websites that 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 collections were done for over a period of twenty months. FAQs were gathered from more than 15 websites,(e.g. http://www.aids.gov/hiv-aids-basics/, https://actgnetwork.org/,http://www.webmd.com/hiv-aids/news/20101215/hiv-aids-cure-faq, http://www.symptomsofhiv.co.za/symptoms-of-hiv.php,http://www.info.gov.za/faq/aids.htm, etc.),literature, books, journal articles, conference proceedings, HIV/AIDS seminars, and workshops, all of which talked about HIV/AIDS-related issues regarding awareness, education, prevention, medications, and therapy.

3. For the SMS normalization process, an electronic version of Collins dictionary was sourced from the web (*http://www.collinslanguage.com/wordlist.aspx.),* and about 40,000 lexicon-type resources were constructed for use in this experimental system for the automated normalization of irregularly-formed English, used in day-to-day communication, in the research domain. This approach is similar to that used for the text normalization objective, where 1,255 entries of a lexical type were gathered in the rule-based approach introduced by Clark and Araki [52].In addition, terms such as abbreviations, acronyms, prepositions, homophones punctuation and medical jargon related to HIV/AIDS were collected as part of the database. Words in the preposition database serve as stop words. Stop words are the name given to words that are altered out prior to, or after, processing of natural language data (text) [39]. Medical jargon was retrieved from different HIV/AIDS websites when FAQ samples were collected. The FAQ document forms a major component of the database used in this research.

5. Keyword extraction algorithm and query code collection

5.1 Keyword extraction algorithm

Keywords in a document provide important information about the content of the document and this helps the users search through information more efficiently [53]. Keywords can be defined as the index terms that contain the most important information for the user. Their purpose is to identify a small set of words from a document that will represent the meaning of the document. Keywords can be stand-alone terms or appear as part of a group of terms with adjacent keywords [54]. They can also be defined as the smallest word unit which expresses the meaning of the entire document, referred to in automatic indexing, text summarization, information retrieval, topic detection and tracking, report generation, web searches, question, and answering, etc. [55]. In text summarization, keywords can be used as a form of semantic metadata [55, 56], beyond content search, index, and rank. Intuitively, the word that appears often in a document but not very often in the corpus is more likely to be a keyword and, conversely, keywords that occur in many documents within the corpus are not likely to be selected as statistically discriminating keyword terms[57]. It is essential that keywords cover the important areas of a document.

This research operates on the individual word in the FAQ documents (i.e. FAQ-query) and FAQ document collection, rather than on a corpus. It can then extract keywords exactly from the FAQ query sentences, regardless of the state of the corpus document. Queries are extracted from the database based on keyword and *n*-gram matching [51]. The use of *n*-grams stands out as an effective tool for the textual computing process over conventional character-based or word-based approaches. As an illustration of their generality, *n*-grams play a role in word-matching, error detection, the correction of spelling errors, string similarity measurement, text retrieval, and searching, language identification and biological sequence computing [58]. An *n*-gram is a substring of length *n* characters derived from a text string; usually, but not necessarily, a word, containing not less than *n* characters. The characters in the *n*-gram retain the same order as in the source text from which the *n*-gram has been derived [58, 59].

The keyword extraction algorithm helps in the FAQ system to identify stop words (or stop lists), phrases, and word delimiters. Candidate keywords are isolated by removing stop words from the FAQ query document collection. What this means is that the word or phrase delimiters will now represent the keyword or key phrase, which are sequences of content words as they occur in the FAQ query document collection. It is on this basis that the scoring function will be calculated. The array of stop words (or stop lists), phrases, and words are split into sequences of contiguous words at phrase delimiters and stop word positions. Every word that is represented in the FAQ files is either a stop word or a candidate keyword, and these categories of words are selected and stored in preposition/punctuation tables in the *MySQL* relational database. In practice, stop lists are often based on common function words and are hand-tuned for particular applications, domains, or specific languages [57].

5.2 Query code identification

The translation of the SMS text to an English form, e.g. *when do you initiate antiretroviral therapy*, is used for this illustration. A new set of SMS queries is formed and this will be used to query the search engine. Query code is essential for easy identification and recognition of each query in the database. It serves the purpose of annotation. Logging data plays a significant role in the evaluation process of a quality search service with a search engine [60] in order to merge data effectively for further data analysis. In Table 2, for interaction purposes, the SMS code (query number) and query code represent the users and the information systems in research communities. For easy identification, each question with its corresponding answers has a unique code. Isolation and identification of the keywords lead to the derivation of further idioms. The interpretation of the wordings is done individually and not collectively. Query code serves as another space to hold the list of the extracted keywords. It is expected that the list of keyword phrase pairs extracted from the query will be randomly or statistically selected terms from the query database and must have been stored in the *MySQL* table. For example, Table 1 could be considered for the generalization of the experiment.

Table 1: Keywords extraction from FAQ data files

SMS code	Query code	Keyword phrase extracted from the query
Q_1	A	$[a_1;:::; a_n]$ list of keywords extracted from A
Q_2	B	$[b_1;:::; b_n]$ list of keywords extracted from B
Q_3	C	$[c_1;:::; c_n]$ list of keywords extracted from C
…	…	…
Q_n	...	...

Table 2: SMS codes, query and keyword extraction

SMS code	Selected keyword phrase extracted from the query
Q_1	When do you initiate antiretroviral therapy?
Q_2	Can HIV be transmitted through breastfeeding?
Q_3	Explain antiretroviral treatment?
Q_4	What are the symptoms of HIV infection?
Q_5	Does breastfeeding pose any risk to the HIV infected mother?
Q_6	What are antiretroviral drugs?
...	...
Q_n	...

There is an average of seven words per question sentence for the FAQ query selected. For each query in the FAQ file there are two things happening: (1) a tag or code is assigned for easy identification, and (2) a list of keyword phrases for every query sentence is created. The underlined words in Table 2 denote the keywords used as references for the query. The parsing rule used for this sample database allowed that keywords may appear in more than one query sentence.

The keyword is coded by assigning token_id in Table 3 by considering the set of keywords $K_1;K_2$; …; Km, acting as the list generated using the keyword extraction algorithm from the FAQ list in Table 2. A token_id is assigned as a whole number from the FAQ query set 1; 2; …, for each keyword. Table 4 illustrates a sample of keywords and their corresponding token_ id in the form of the vector space.

Table 3: Assigning token_id to the keyword

Token_id	Keywords
K_1	Initiate
K_2	Antiretroviral
K_3	Therapy
K_4	Drugs
K_5	Transmitted
K_6	Breastfeeding
K_7	Symptom
...	...
K_m	…

Table 4: (n x m) term-document matrix corresponding to the FAQ sentences

Token_ids	Q_1	Q_2	Q_3	Q_4	Q_5	…	Q_n
K_1	0	0	0	0	0	…	0
K_2	0	0	0	0	0	…	0
K_3	0	0	0	0	0	…	0
K_4	0	0	0	0	0	…	0
…	…	…	…	…	…	…	…
K_m	…	…	…	…	…	…	…

Corresponding to the text in Table 2 is the n×m term dependent matrix shown in Table4. The elements of this matrix are the frequencies with which a term occurs in the FAQfile. This is used for the scoring function. The scoring function is the addition of the weighting in each query column. The results are ranked to give a list of the query-answer pair. Using SMS codes Q_6 in Table 2—*What are antiretroviral drugs*—for illustration, the contents of the seventh column in the n×m term-document matrix, *antiretroviral* and *drugs*, all occur once. A value of 1 is assigned to the term if it is available, otherwise 0. Thetoken_ids of *antiretroviral* and *drugs* are K_2 and K_4 respectively.

The query set, *Q*, is represented as $Q_1;Q_2;\ldots;Q_n$ in Table 4. The term-document matrix table is used to calculate the frequency of keyword $K_1;K_2;\ \ldots,K_n$, in the query sentence *Q*. The corresponding values of *K* in *Q* may be Boolean values, depending on whether it is present or not in the query sentence.

The schema of FAQ document has three columns, as shown in Table 5: (1) Qcode–{a unique auto-incremental key that serves as the primary key (PK) for easy identification of the query and the answer pair; (2) Query–{this attribute has a list of 350 FAQs within the domain of studies (medical); and (3) Answer–{this attribute contains the answers to each query.

Table 5: MySQL description of the FAQ database table

Field	Type	Key	Default	Extra
Qcode	Int (255)	Primary	Null	Auto Increment
Query	Varchar (100)	-	-	-
Answer	Varchar (100)	-	-	-

Table 6: Relevance judgment value

Relevance judgment	Value
Excellent	5.0
Very good	4.0
Good	3.0
Moderate	2.0
Poor	1.0

The database structure for the FAQ information retrieval system has one table with 350 HIV/AIDS queries. MySQL, a relational database, was used to store FAQ and answers datasets for future data analysis. The primary objective of this evaluation is to compare the retrieval performance in the experiments using these algorithms: *naïve query retrieval*, *tf-idf*, and *SMSql* (an algorithm the researcher has developed).

6. Research methodology

The efficiency of the retrieval mechanism is determined by its performance. The best retrieval strategy may depend greatly on the length and specificity of the query, because a complex data-driven retrieval strategy may have little success with short queries and limited amounts of information[61]. Users of search engines have been accustomed to using short queries

with keyword combinations due to the interface restrictions and inner mechanism of the search engine[61]. However, the detail that they provide may 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 [62].
As a result, retrieval efficiency can be calculated through precision, recalls and f-measure. The learning performance involves performing the same set of experiments with a predetermined number of iterations on the same dataset a particular number of times. To conduct the evaluation, the following steps are taken:
1. A sample of twenty (20) SMS coded FAQ query sentences were taken. (*Mostly they are a set of queries that have greater representation in the data collected from the respondents. This has been determined statistically*)
2. Each query was designed to retrieve the five (5) best answers. The results will be verified by experienced users, using the experimental datasets applied at the beginning of the experiment, and their corresponding answers.
3. The retrieval efficiency was measured using *precision* and *recalls*.
4. The retrieval time of the three algorithms was compared in order to know the fastest algorithm.
Precision is the relative amount of correct constituent (FAQ query) retrieved out of 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 [35].
Precision = Number of relevant FAQ query / Number of retrieved FAQ query
Recall is the relative number of correct constituents compared to the gold standard parse. It shows how many relevant answers were actually retrieved out of the possible answers, the higher the recall value, the better the algorithm performance.

The two metrics, *precision* and *recall*, which are inversely related are computed using the unordered list of FAQ query sets [63]. They are based on the user's relevance assessments following the retrieval process [62]. Therefore, the automatic handling of the various forms of user queries not only requires a large database of QA pairs but also the technology to match the user query to the FAQ documents in the database [37]. It is imperative to link information seekers to information sources by matching the SMS query with the description of the content that is associated with the indexed information segments in the database. The FAQ dataset comprised English words and HIV/AIDS terminologies.

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 which are relevant to each query have already been statistically identified. The performance of each automatic system can then be compared to a known standard of optimal performance. Systems are rated according to their ability to rank the relevant documents higher than the documents which are not relevant. While one can give a number of arguments about how and why this test set does not reflect reality, no better methods for evaluating performance have been developed [64].

Table 5shows the relevance judgment scale needed to calculate retrieval efficiency. The judgment is based on the first 5 FAQ sets of queries that emerge from various ways in which SMS questions are sent into the search engine. This approach is similar to Mogadala et al.[45], where cleaned SMS was used as a query to match the 5 best documents containing FAQ questions, 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 compared. The mapping will assist in determining the best retrieval efficiency of the three algorithms. A maximum of 5 points is allotted to an SMS enquiry that exactly produces the intention of the SMS texter in terms of the FAQ data set. A value of 0-point may be considered for out-of-domain situations where the result of the FAQ query has no bearing on the SMS enquiry. Some SMS queries will be out-of-domain and will not have any corresponding FAQ answers[19, 45].

6.1 Scoring function

Average precision and *average recall* were the means of the *precision* and *recall* values obtained respectively, from the set of the 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 if the set of relevant FAQ query documents for an information need$q_j \in Q$ is $\{d_1, \dots, d_{m_j}\}$ andR_{jk} is the set of ranked retrieval results from the top result until you get to the FAQ query document d_k.
When a FAQ file is chosen as the query is being issued, the system iterates through the QA pairs in the file, comparing each question against the user's question and computes a score based on the *weight function*. We define a *scoring function* for assigning a score to each statistically selected keyword phrase in the equation corpus Q, where SMS token s_i has been

normalized to English term t in the dictionary. Therefore, there is a similarity measure §, between s_i and t such that $\S(s_i, t) > 0$ and this is denoted in the equation as $s_i \sim t$ in the equation. The *score function* measures how closely the question matches 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* chooses the term from q having the maximum weight. Then the weight n of the chosen terms are summed together, this gives the score

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

The goal is to efficiently find the best matches to the query in the FAQ. The five selections with the highest scores are selected and are returned to the user. Each question from the FAQ file is matched against the user's question and then scored.

7. The algorithms

There are three algorithms used in this research to confirm the research questions

A. SMS question locator (*SMSql*) (the proposed algorithm)

This section discusses the *SMSql* algorithm over the SMS FAQ search and retrieval system for mobile communication. The translated keywords extracted from the SMS-query are matched with words present in our FAQ corpus. The algorithm considered similarity in words between the normalized SMS query sentence from the mobile user and the FAQ database, the length of the two sentences as well as the order in which the words are placed. The length of the query sentence is given priority in assigning the weight function.

One of the methods adopted in arriving at a ranked list is assigning weights to the relevant terms. This shows the degree of importance of the terms (tokens) in the documents. The weight difference is needed for the following reasons: (1) to measure the degree of similarity between the FAQ terms and SMS query terms. (2) to know the length and specificity of the query sentences, and the number of relevant FAQ documents (sentences) i.e. the number of query sentence terms. A weight function/value of 1 is to measure the similarity between the FAQ terms and the SMS query terms. A weight function/value of 2 is used to confirm the FAQ query sentence length. For as many keyword terms that are available in the FAQ sentence (and non-matching) are assigned 2. This is important if there is a tie in the weight function between FAQ terms and SMS query. The FAQ query sentence with lower sum of non-matching terms is considered as the chosen FAQ query sentence.

SMSql processes the input sentence word-by-word from left to right. When the first SMS word (target word) is found, the context window is built. This window is 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. [65] and Michelizzi [66] that words farther away from the target word are less likely to be related to words close to the target word. Figure 2 illustrates the step-by-step description of the *SMSql* algorithm.

A. SMSql 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:Step-by-step description of the *SMSql* algorithm.

B. 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 twenty questions. This is ranked and represents relevance of the questions based on the SMS enquiries for this approach. *Figure 3* illustrates the step-by-step description of the *tf-idf* algorithm.

Tf-idf 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, corrected, correcting, corrects.*
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: Step-by-step description of the *tf-idf* algorithm.

C. Naive (Brute-force string match) algorithm

This problem involves searching for a pattern (substring) in a string of text. The result is either the index in the text of the first occurrence of the pattern or indices of all occurrences. We will look only for the first match. Naive retrieval is done by brute force whereby the list of queries is traversed to count the frequency of occurrences of a particular word [49].*Figure 4* illustrates the step-by-step description of the *naive* algorithm.

Naive algorithm

Step 1	*Align the pattern at beginning of the text*
Step 2	*Moving from left to right, compare each character of the pattern with the corresponding character in the text until all characters are found to match (successful search) or a mismatch is detected*
Step 3	*While pattern is not found and the text is not yet exhausted, realign the pattern one position to the right and repeat Step 2*

Figure 4: Step-by-step description of the *naive* algorithm.

8 Results

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

8.1 Average precision and average recall for the three algorithms

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 normalized SMS queries. The peak of the graph is where the most relevant of the query is fetched. The results are the mean values for each SMS query for the three algorithms we are considering.

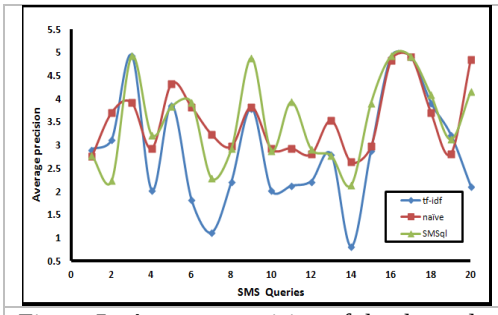

Figure 5. Average precision of the three algorithms

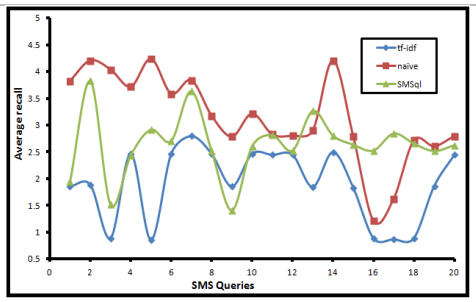

Figure 6. Average recall of the three algorithms

Overall, the performance of SMS queries in the three algorithms appeared very difficult to confirm. Thus there was a need to perform a statistical test on the results and confirm the significant test. A significance test was adopted to reject the null hypothesis, H_0, that there is no difference between the results of the three algorithms. The *one way* repeated measure

ANOVA was used because of each method (algorithm) is exposed to three conditions (*precision*, *recall*, and *timing*) [67]. There is continuous scaling on the three methods.

The descriptive analysis of the three methods using *average precision statistical analysis*, where 20 query samples (*N*) were used. The results of the SMSql gave the best Mean (3.55) and a moderate standard deviation (0.999). The *one way* repeated measures ANOVA was conducted to compare the confidence interval of the three algorithms with the same set of queries. The mean and standard deviation are presented gives *tf-idf* (mean=2.90, std. deviation=1.252); *naive* (mean=3.52, std. deviation=0.786); and *SMSql*(mean=3.55, std. deviation=0.999). There was a significant effect in the result of *SMSql* algorithms (Wilks' Lambda =.59; F(2,18) = 6.261; p <.005, multivariate partial eta squared = .1).

The descriptive analysis of the three methods on *average timing statistical analysis*, where 20 sample queries (N) were used. The results of the *SMSql* gave the best Mean (0.70) and a moderate standard deviation (0.013).The *one-way* repeated measure, ANOVA, was conducted to compare the confidence interval of the three algorithms with the same set of queries. The mean and standard deviations are presented gives *tf-idf* (mean=0.073, std. deviation=0.0113); *naive* (mean=0.078, std. deviation=0.0079); and *SMSql* (mean=0.0698, std. deviation=0.0129). There was a significant effect in the result of *SMSql* algorithms (Wilks' Lambda =.58; F(2,18) = 6.444; p < .005, multivariate partial eta squared = .42).

8.2 Average retrieval time of the three algorithms

The evaluation is carried out by computing the time taken for the retrieval of documents in the FAQ system. In order to demonstrate clearly the effectiveness of each method, the selection of a feature set from data showing high statistical dependencies provides a more discriminating test [68]. The execution time to generate results was compared for the three algorithms. The system of calculating execution time can be constructed out of sequential programs but are typically built from concurrent programs called tasks [69]. The average time taken t_n for a query Q_n for each SMS request by the user is taken for each of the algorithms, and the results are presented in Table 7. The score of each query sentence is calculated sequentially and then ordered to generate the result. The average time for each iteration of the SMS queries, $Q_1 \rightarrow Q_{20}$, for each algorithm was taken. The results show the time spent in generating responses to requests made in this experiment.

Table 7. Time computation for the retrieval process of the SMS queries

SMS query no.	Average computational time for each iteration of the SMS query		
	SMSql (sec)	Tf-idf (sec)	Naive(sec)
1	0.097	0.085	0.099
2	0.087	0.077	0.082
3	0.076	0.086	0.076
4	0.072	0.074	0.068
5	0.085	0.085	0.078
6	0.037	0.037	0.065
7	0.069	0.069	0.075
8	0.067	0.067	0.072
9	0.072	0.077	0.080
10	0.068	0.068	0.072
11	0.074	0.074	0.074
12	0.078	0.088	0.090
13	0.057	0.067	0.077
14	0.075	0.075	0.079
15	0.062	0.082	0.089
16	0.057	0.067	0.077
17	0.078	0.078	0.082
18	0.062	0.068	0.078
19	0.059	0.059	0.069
20	0.064	0.074	0.080
Total	1.396	1.457	1.562
Average	0.070	0.073	0.078

The retrieval speed improvement of the *tf-idf* and *naïve* algorithms in relation to the proposed algorithm *SMSql,* is (1) *tf-idf* and *SMSql:* $\frac{0.073 - 0.070}{0.073} \times 100\% = 4.1\%$ and (2) *naive* and *SMSql:* $\frac{0.078 - 0.070}{0.078} \times 100\% = 10.3\%$. The results show 4% and 10.3% improvement in the retrieval efficiency measured by computational speed between *SMSql* and the other methods (*tf-idf* and *naïve*) respectively. *SMSql* was the fastest.

9. Conclusion and further work

This paper investigates the use of SMS in seeking information on health-related matters. A pre-processing stage, i.e. SMS normalization is important before it can be used for retrieval processes. The normalized SMS terms are used in the form of the vector-space-model to retrieve answers from a FAQ database using the developed retrieval algorithm. The best-matching SMS normalized and FAQ terms can be found by processing just those normalized SMS lists that are associated with the *n*-grams comprising the query word for which the keyword variants are required. The results were compared using three retrieval metrics (precision, recall and retrieval speed). There were significant effects of the developed algorithm with all the metrics compared to other methods.

Funding

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

Acknowledgment

Professor Isabella M. Venter is appreciated for her support in the course of the research.

References

[1] R. A. Atun, S. R. Sittampalam, and A. Mohan, "Uses and benefits of SMS in healthcare delivery," *Discussion paper. London: Imperial College,*, 2005.

[2] M. H. van Velthoven, L. T. Car, J. Car, and R. Atun, "Telephone consultation for improving health of people living with or at risk of HIV: a systematic review," *PloS one*, vol. 7, p. e36105, 2012.

[3] M. H. van Velthoven, L. Tudor Car, and J. Car, "Telephone communication of HIV testing results for improving knowledge of HIV infection status," *The Cochrane Library*, 2011.

[4] V. Ostojic, B. Cvoriscec, S. Ostojic, D. Reznikoff, A. Stipic-Markovic, and Z. Tudjman, "Improving asthma control through telemedicine: a study of short-message service," *Telemed J E Health*, vol. 11, 2005.

[5] D. West, "How mobile devices are transforming healthcare," *Issues in technology innovation*, 2012.

[6] J. Nesbit, "Building an SMS Network into a Rural Healthcare System," 2011.

[7] S. Bergvik and R. Wynn, "The use of short message service (SMS) among hospitalized coronary patients," *General hospital psychiatry*, vol. 34, pp. 390-397, 2012.

[8] C. Fairon and S. Paumier, "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.

[9] F. Alam, S. Habib, and M. Khan, "Text normalization system for Bangla," BRAC University2008.

[10] R. Beaufort, S. Roekhaut, L.-A. Cougnon, and C. Fairon, "A hybrid rule/model-based finite-state framework for narmalizing SMS messages," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguists*, Uppsala, Sweden, 2010, pp. 770-779.

[11] P. Deepak and V. Subramaniam, "Correcting SMS Text Automatically," 2012.

[12] E. Thuma, S. Rogers, and I. Ounis, "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.

[13] B. J. Jansen and U. Pooch, "A review of web searching studies and a framework for future research," *Journal of the American Society for Information Science and Technology*, vol. 52, pp. 235-246, 2001.

[14] D. Lewandowski, "Search engine user behaviour: How can users be guided to quality content?," *Information Services and Use*, vol. 28, pp. 261–268, 2008.

[15] A. Moreo, M. Romero, J. L. Castro, and Zurita J.M., "FAQtory: A framework to provide high-quality FAQ retrieval systems," *Expert Systems with Applications*, 2012.

[16] J. Leveling, "On the Effect of Stopword Removal for SMS-Based FAQ Retrieval," in *Natural Language Processing and Information Systems*. vol. 7337, G. Bouma, A. Ittoo, E. Métais, and H. Wortmann, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 128-139.

[17] J. Leveling, "DCU@ FIRE 2012: Monolingual and Crosslingual SMS-based FAQ Retrieval," 2012.

[18] G. Anderson, S. D. Asare, Y. Ayalew, D. Garg, B. Gopolang, A. Masizana-Katongo, *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.

[19] D. Hogan, J. Leveling, H. Wang, P. Ferguson, and C. Gurrin, "DCU@ FIRE 2011: SMS-based FAQ Retrieval," in *3rd Workshop of the Forum for Information Retrieval Evaluation, FIRE*, 2011, pp. 2-4.

[20] R. Sproat, A. Black, S. Chen, S. Kumar, M. Ostendorf, and C. Richards, "Normalization from non-standard words," *Computer speech and language*, vol. 15, pp. 287-333, 2001.

[21] A. Aw, M. Zhang, P. Yeo, Z. Fan, and J. Su, "Input Normalization for an English-to-Chinese SMS Translation System," *MT Summit-2005* 2005.

[22] Z. Xue, D. Yin, and B. D. Davison, "Normalizing Microtext," presented at the Proceedings of the AAAI-11 Workshop on Analyzing Microtext: San Francisco, USA Department of Computer Science & Engineering, Lehigh University Bethlehem, PA 18015 USA, 2011.

[23] J. Chen, B. Linn, and L. Subramanian, "SMS-based contextual web search," in *Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*, 2009, pp. 19-24.

[24] J. Chen, L. Subramanian, and E. Brewer, "SMS-based mobile web search for low-end phones.," presented at the 16th Annual International Conference on Mobile Computing and Networking, ACM, 2010.

[25] P. Cook and S. Stevenson, "An unsupervised model for text message normalization," in *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity*, Boulder, Colorado, 2009, pp. 71–78.

[26] S. Acharyya, S. Negi, L. Subramaniam, and S. Roy, "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*, 2008, pp. 67–74.

[27] D. Vilariño, D. Pinto, B. Beltrán, S. León, E. Castillo, and M. Tovar, "A Machine-Translation Method for Normalization of SMS," in *Pattern Recognition*. vol. 7329, J. Carrasco-Ochoa, J. Martínez-Trinidad, J. Olvera López, and K. Boyer, Eds., ed: Springer Berlin Heidelberg, 2012, pp. 293-302.

[28] P. Koehn, "Europarl: A parallel corpus for statistical machine translation," in *MT summit*, 2005.

[29] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, Edmonton, Canada, 2003, pp. 48–54.

[30] L. Lee, "Measures of distributional similarity," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, 1999, pp. 25–32.

[31] D. A. Jackson, K. M. Somers, and H. H. Harvey, "Similarity coefficients: measures of co-occurrence and association or simply measures of occurrence?," *American Naturalist*, pp. 436–453, 1989.

[32] P. Pakray, S. Pal, S. Poria, S. Bandyopadhyay, and A. Gelbukh, "SMSFR: SMS-Based FAQ Retrieval System," in *Advances in Computational Intelligence*, ed: Springer, 2013, pp. 36-45.

[33] Z. Elberrichi, A. Rahmoun, and M. A. Bentaallah, "Using WordNet for Text Categorization," *Int. Arab J. Inf. Technol.*, vol. 5, pp. 16–24, 2008.

[34] G. Anderson, Y. Ayalew, P. A. Mokotedi, N. P. Motlogelwa, D. Mpoeleng, and E. Thuma, "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, pp. 307 - 313.

[35] A. Masizana-Katongo and T. Ama-Njoku, "Example-Based Parsing Solution for a HIV and AIDS FAQ System," *International Journal of Research and Reviews in Wireless Communications (IJRRWC)*, vol. 1, pp. 59-65, 2011.

[36] G. Kothari, S. Negi, T. A. Faruquie, V. T. Chakaravarthy, and L. V. Subramaniam, "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.

[37] Akhil Langer, Rohit Banga, Ankush Mittal, and L.V. Subramaniam, "Variant Search and Syntactic Tree Similarity Based Approach to Retrieve Matching Questions for SMS queries," presented at the AND'10, 2010.

[38] M. Jain, "N-Gram Driven SMS Based FAQ Retrieval System," MSc, Computer engineering, Delhi College of Engineering Delhi University, Delhi, 2012.

[39] A. Joshi, "Improving accuracy of SMS based FAQ retrieval," *International Journal of Emerging Technologies in Computational and Applied Sciences*, pp. 362-366, 2012.

[40] W. Y. Conwell, "Methods and systems for content processing," ed: Google Patents, 2012.

[41] M. Badawi, A. Mohamed, A. Hussein, and M. Gheith, "Maintaining the search engine freshness using mobile agent," *Egyptian Informatics Journal*, 2012.

[42] Y. Arens, C. A. Knoblock, and W.-M. Shen, *Query reformulation for dynamic information integration*: Springer, 1996.

[43] P. Boldi, F. Bonchi, C. Castillo, and S. Vigna, "Query reformulation mining: models, patterns, and applications," *Information retrieval*, vol. 14, pp. 257-289, 2011.

[44] K. Komiya, Y. Abe, H. Morita, and Y. Kotani, "Question answering system using Q & A site corpus query expansion and answer candidate evaluation," 2013.

[45] A. Mogadala, K. Rambhoopal, and V. Varma, "Language modeling approach to retrieval for SMS and FAQ matching," 2012.

[46] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic, "Real life information retrieval: a study of user queries on the Web," in *ACM SIGIR Forum*, 1998, pp. 5-17.

[47] J. Ramos, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the First Instructional Conference on Machine Learning*, 2003.

[48] R. D. Burke, K. J. Hammond, V. Kulyukin, S. L. Lytinen, N. Tomuro, and S. Schoenberg, "Question answering from frequently asked question files: Experiences with the faq finder system," *AI magazine*, vol. 18, p. 57, 1997.

[49] B. Kaur, A. Saxena, and S. Singh, "Web opinion mining for social networking sites," in *Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology*, 2012, pp. 598-605.

[50] Ipoletse, "Ipoletse Training Manual for Call Centre Operators for the National Call Centre on HIV AND AIDS, Gaborone, Botswana," ed, 2002.

[51] A. Masizana-Katongo, G. Anderson, D. Mpoeleng, T. Taukobong, G. Mosweunyane, O. T. Eyitayo, *et al.*, "An SMS-based Healthcare Information Storage and Retrieval System," in *Proceeding of the IASTED African Conference Health Informatics (AfricaHI2010)*, Gaborone, Botswana 2010.

[52] E. Clark and K. Araki, "Two Database Resources for Processing Social Media English Text," in *Proceedings of the 31st Annual Meeting on Association for Computational Linguistics, LREC '12* 2012, pp. 3790–3793.

[53] F. Liu, D. Pennell, F. Liu, and Y. Liu, "Unsupervised approaches for automatic keyword extraction using meeting transcripts," in *Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics*, 2009, pp. 620-628.

[54] L. Wang and D. Wang, "Method for ranking and sorting electronic documents in a search result list based on relevance," ed: WO Patent WO/2007/089,289, 2007.

[55] R. Barzilay and M. Elhadad, "Using Lexical Chains for Text Summarization," in *Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain, 1997, pp. 10-17.

[56] E. D'Avanzo and M. Bernardo, "A Keyphrase-Based Approach to Summarization: the LAKE System at DUC-2005," presented at the Document Understanding Conference, 2005, 2005.

[57] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic keyword extraction from individual documents," *Text Mining*, pp. 1-20, 2010.

[58] A. M. Robertson and P. Willett, "Applications of-grams in textual information systems," *Journal of Documentation*, vol. 54, pp. 48-67, 1998.

[59] L. Egghe, "The distribution of N-grams," *Scientometrics*, vol. 47, pp. 237-252, 2000.

[60] T. Mandl, M. Agosti, G. M. Di Nunzio, A. Yeh, I. Mani, C. Doran, *et al.*, "LogCLEF 2009: the CLEF 2009 multilingual logfile analysis track overview," in *Multilingual Information Access Evaluation I. Text Retrieval Experiments*, ed: Springer, 2010, pp. 508–517.

[61] D. Wu, Y. Zhang, S. Zhao, and T. Liu, "Identification of Web Query Intent Based on Query Text and Web Knowledge," in *Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on*, 2010, pp. 128-131.

[62] S. Maleki-Dizaji, "Evolutionary Learning Multi-Agent Based Infromation retrieval Systems," *PhD Thesis Sheffield Hallam University*, 2003.

[63] M. Buckland and F. Gey, "The relationship between recall and precision," *JASIS*, vol. 45, pp. 12–19, 1994.

[64] D. A. Hull, "Information retrieval using statistical classification," PhD Thesis, Department of Statistics, Stanford University, 1994.

[65] L. B. Huang, V. Balakrishnan, and R. G. Raj, "Improving the relevancy of document search using the multi-term adjacency keyword-order model," *Malaysian Journal of Computer Science*, vol. 25, p. 1, 2012.
[66] J. Michelizzi, "Semantic relatedness applied to all words sense disambiguation," University of Minnesota, 2005.
[67] J. Pallant, *SPSS survival manual: A step by step guide to data analysis using SPSS*: Open University Press, 2010.
[68] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," *Pattern recognition letters*, vol. 15, pp. 1119-1125, 1994.
[69] P. Puschner and C. Koza, "Calculating the maximum execution time of real-time programs," *Real-time systems*, vol. 1, pp. 159-176, 1989.