A Mobile-Health Information Access System

Ademola O. Adesina and Henry O. Nyongesa
Department of Computer Science
University of the Western Cape
Bellville, Cape Town, South Africa
Email: {inadesina, henrynyongesa}@gmail.com
Telephone: (021) 959–3010, Fax: (021) 959–3006

***Abstract*—Patients using the Mobile-Health Information System can send SMS requests to a Frequently Asked Questions (FAQ) web server with the expectation of receiving an appropriate feedback on issues that relate to their health. The accuracy of such feedback is paramount to the mobile search user. However, automating SMS-based information search and retrieval poses significant challenges because of the inherent noise in SMS communication. First, in this paper an architecture is proposed for the implementation of the retrieval process, and second, an algorithm is developed for the best-ranked question-answer pair retrieval. We present an algorithm that assists in the selection of the best FAQ-query after the ranking of the query-answer pair. Results are generated based on the ranking of the FAQ-query. Our algorithm gives a better result in terms of average precision and recall when compared with the *naïve* retrieval algorithm.**

***Index Terms*—Information Retrieval, SMS/Text Message, Frequently Asked Question (FAQ), HIV/AIDS, mHealth, Question Answer (QA)**

I. Introduction

Social networking communication such as Facebook, Instant Messaging, MXit, Short Message Services (SMS) and Twitter produce vast amounts of noisy text data that require some form of cleaning before such data can be useful for information processing tasks, such as Frequently Asked Question (FAQ) answering system. SMS has become a common way of communication as a result of the explosive use of mobile communication all over the world. It is widely accepted especially among the youth, because of its flexibility in the use of alphanumeric characters, with little or no regard for orthographical and grammatical rules. This computer-mediated communication has its own peculiarities, where groups of users have their own patterns of writing, inventing new abbreviations, and using the non-standard orthographic forms [1]. The freedom of writing poses a great challenge to its transformation into the formal writing suitable for information processing. SMS language has, however, been recognized and accepted as a variant of natural languages [2]. Thus, there is a compelling motivation to build many information-based services around the SMS communication through the process of normalizing its various forms in which the language appears [3], [4].

With regard to mobile information search and retrieval, the length of time spent by a mobile user at a particular search service is usually brief because the answer retrieved may be un-/satisfactory or un-/available. The eagerness of mobile searchers tends to vary, because they approach the search engine with a specific topic in mind and their search often does not lead to results that satisfy their needs, unlike in desktop search [5]. There is a limitation to the level at which information is made available in a mobile information retrieval system. This may be attributed to the restriction in the bandwidth, size of the keypad, or the restriction in the SMS size constrained to 140 bytes [6], [7].

In mobile health (mHealth) technology SMS has played a significant role in bridging the gap of communication between the patients and health workers. It frees the physicians from routine office visits while still providing consultancy to patient's complaints and conditions, so creating time for patients that require more detailed medical attention. Mobile health activities have sprung up as health call centres which respond to patient enquiries [8], [9]. SMS is a form of the mobile technological approach that is used to reach the patient for drug administration, consultancy services, appointment reminders, health and prevention reports, billing information and so on [10], [11]. The role of SMS in mHealth services cannot be underestimated, for instance, it is used in South Africa to remind tuberculosis 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 [8].

The paper is organized as follows. Section 2 presents related works on SMS-based information retrieval system, while Section 3 introduces the system architecture in building the SMS-query system. Section 4 discusses the research methodology adopted in building the SMS-based FAQ system and its implementation in an information retrieval (IR) system. Section 5 presents the proposed algorithm *SMSql* for FAQ search and retrieve. The performance evaluation and the metric indices are discussed in Section 6. In Section 7, the comparison of the results from *naïve* and *SMSql* algorithms are discussed. The conclusion of the paper is given in Section 8.

II. Related work

Significant work has been done on SMS-based FAQ systems. Different attempts with distinct contexts have been made to normalize text messages. In this paper, we focus on the use of normalized SMS for information retrieval from the

information source. Hogan et al. [3] described SMS-based FAQ retrieval systems as having three stages: (1) SMS normalization, (2) retrieval of ranked results and (3) identifying out-of-domain query results. The SMS FAQ queries were manually annotated from micro-text corpora. The tokens were aligned with the original text messages to give one-to-one correspondence between the original and corrected tokens. The documents and SMS questions underwent the same pre-processing of annotation. 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 two search engines.

SMSFind is another SMS-based information retrieval model proposed by Chen et al. [6]. It uses the conventional search engine in the back-end to provide an appropriate answer for the SMS request. *SMSFind* uses the translated 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 SMS 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* list for each page and it is ranked altogether. The highest score is released as an answer to the request [6]. The use of *hint*s in the algorithm is considered a supervised learning approach [12], [13] and it adds costs to generate and store. The research never considered the contextual information of the searches. The searching is limited to the constituent of the *hint*.

SMSFR is a recent SMS-based searching technique developed by Pakray et al. [14]. It has a multi-lingual (English, Hindi and Malayalam) feature with multi domain FAQ datasets. 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 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. The WordNet is a lexical database for the English language that groups English words into sets of synonyms called synsets [15]. 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 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 SMS query 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 a free software.

Healthcare FAQ information retrieval systems using SMS in form of a Question and Answer (Q&A) System were recently proposed by Anderson et al. [16] and Masizana-Katongo et al. [17]. SMS users submit queries to the portal through a mobile phone interface. A *parsing technique* was proposed as a 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 by 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* format into pre-processed parsed tree building blocks that can be easily implemented in Java code. The system is evaluated using recall, precision and rejection. Their procedure did not consider ranking of the SMS query in presenting the answer.

Kothari et al. [19] designed an automatic FAQ-based question answering system. The method involves determining SMS-query similarity over 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 problem 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 the information retrieval efficiency. The model involves the use of a dictionary, and maps the SMS query to the questions in the corpus. However, the noise removal step is computationally expensive [20]. Kothari et al.'s [19] system does not involve training data nor normalization. It has the advantage of handling semantic variations in question formulation but the method fails to discuss the choice of homophonic words as regards automatic speech recognition.

In our previous work the method of SMS normalization was

discussed [21]. The cleaned SMS-based query is used to query the FAQ database system. Anderson et al. [16] and Masizana-Katongo et al. [22] 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 multi-lingual scenario whereas we considered English only. The system architecture for our proposed SMS-based information retrieval system is discussed in the next section.

III. System architecture

Figure 1 shows how the SMS query is presented to the web search engine. A normalized SMS is made to interface with the QA database. The web server contains the FAQ-SMS database, predefined queries, and corresponding answers to the queries. The set of query documents 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 [23], [24]. 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. 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.

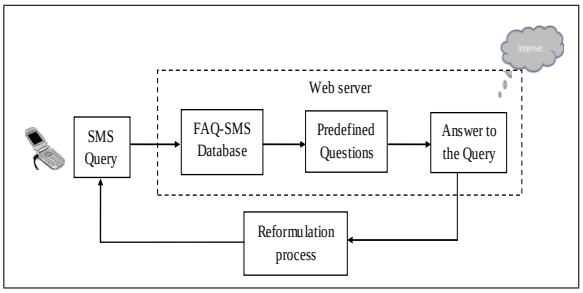

Fig. 1. System architecture of an SMS-Query and reformulation process

The methodology adopted to access information using cell-phones is done with our 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. Communication is triggered by the SMS sent by the user and received by the web server on the system. A preliminary process translates the wrangled SMS text to its English form and then the noise-free query is parsed using the SMS parser (SCORE algorithm) as discussed in our previous work, Adesina et al. [21].

IV. RESEARCH METHODOLOGY

There are various ways of collecting datasets for an experiment. For instance, the experiment performed by Jansen et al. [25] 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 using *tf-idf* to determine word relevance in document queries. From the document, 86 queries were extracted to perform the experiment on information retrieval [26]. A widely read and popular news media from the blog of *The Times of India,* was used as the source of data. The blog has several datasets on topics covering politics, sports, entertainment, cuisine, social evils [27]. For our experiment, a FAQ database consisting of over 350 sampled questions was built from websites and data from research communities. Our focus centres on HIV/AIDS drug administration, prevention, control and support, counselling, food prescription, awareness, sex education, and education and training. Out of the 350 sampled FAQ collections, 200 questions were extracted from the Ipoletse call centre manual [28] and the remaining were fetched from related websites. The Ipoletse database consists of most frequently asked questions about HIV/AIDS and ARV therapy, the booklet was prepared by the Ministry of Health in Botswana. These websites have vast information on the HIV/AIDS epidemic in the form of answers to FAQs on aspects of drug administration, therapy, sex education, food and nutrition, physical exercise and treatment. The FAQ dataset comprises English words and HIV/AIDS terminology. The online collections were done for over twenty months.

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 the data analysis. The MySql description of the FAQ database table is as shown in *Table 1*. The schema has three columns—(1) *Qcode*– is a unique auto-incremental key that serves as the primary key (PK) for ease 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 I
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)	-	-	-

The research motive is to compare the retrieval performance of the—*Naive query retrieval* and *SMSql*, a developed algorithm, when SMS is used as the query.

From the FAQ query set, 20 questions were used for the analysis. These sets of questions were translated to SMS shorthand by students of the University of the Western Cape. A set of 20 questions from 100 respondents yielded 2,000 SMS query formats used in our dataset, that is, each query had 100 respondents. A large collection of data was necessary in order to reduce the tendency of bias in the SMS writing. Extracting the best matching question-answer pairs in the server is our ultimate goal. This is achievable by statistically selecting *keywords* and *idioms* from the query corpus in the FAQ query-set gathered earlier. The keywords and idioms are a combination of words or phrases that give a reasonable meaning to each query. From the keyword phrases, idioms

can be derived; an idiom 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 [29]. Keywords for all the questions are extracted based on the frequency level from over 100 respondents. In summary, 205 keywords were extracted from the FAQ collections. This means that each question has an average of three keyword terms.

At this stage it is important to note that stop words are less important parts of the keyword phrases and are discarded. Stop words are very common words that appear frequently in text and carry little or no semantic meaning in an expression [30], [31]. 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 [32], [33]. It is therefore recommended that high frequency word n-grams that occur in many words will have to be eliminated before computing the similarity coefficient. An n-gram is a contiguous sequence of n items from a given sequence of text or speech [34]. 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 [35].

The retrieval efficiency results of the two algorithms—*naïve* and *SMSql*—and the accuracy of the FAQ question-answer pair returned are used as the basis for judging the efficiency of the algorithm. The relevance judgment needed to calculate the retrieval efficiency is placed on a scale of 5, where excellent $= 5$; very good $= 4$; good $= 3$; moderate $= 2$; and poor $= 1$. The judgment is based on the first 5 FAQ queries that emerge from various ways in which SMS questions are sent into the search engine. This approach is similar to that of Mogadala et al.'s [36] method where a cleaned SMS was used as a query to the search engine. A set of 5 best documents containing FAQ question-and-answer pair emerged as the results, using the language model approach. A maximum of 5 point will be allotted to an SMS enquiry that exactly produces the expectation of the SMS texter in terms of the FAQ data set. A score of 0 point is given for of out-of-domain queries whereby the result of the FAQ query is completely different from the SMS enquiry. Some SMS queries will be out-of-domain and will not have any corresponding FAQ answer [3], [36]. The next section will present the two algorithms used in the SMS-based information accessing techniques.

V. THE SMS QUESTION LOCATOR (*SMSql*) ALGORITHM

This section describes the *SMSql* algorithm used in 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 corpus.

Our algorithm considered similarity in words between the SMS query and the FAQ database, the sentence length of the two sentences as well as the order in which the words are placed. This is taken as an enhancement in our algorithm because length of the query sentence is given priority. For easy identification, each question with its corresponding answers has a unique code. Isolation and identification of the keywords lead to further derivation of idioms.

The *SMSql* algorithm is described next.

A. SMSql algorithm

The *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

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 of 3 was used in our system, which included the target word and one word to its left and right, following the claim by Michelizzi [37] that words farther away from the target word are less likely to be related to words close to the target word.

When a FAQ question 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 question corpus Q, where SMS token s_i has been normalized to the English term t in the dictionary. Their similarity measure *Sim,* is calculated such that $Sim(s_i, t) > 0$ and this is denoted in the equation as $s_i \approx t$. The *score function* measures how closely the question matches the SMS question string S.

Consider a query term $q \in Q$ a FAQ dataset for each token SMS string s_i, the *scoring function* chooses the term q having the maximum weight. Then the weight of the chosen terms are summed together, to give the score.

$$\text{score}(q) = \sum_{i=1}^{n} \max_{t \in Q \wedge s_i \approx t} \big(w(s_i, t)\big)$$

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

B. Naïve (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. The algorithm follows

The *Naïve* 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

VI. PERFORMANCE EVALUATION

A simple 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 which performs optimally. Systems are rated according to their ability to rank relevant documents higher than those documents which 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 [38].

The efficiency of the retrieval mechanism is determined by the system retrieval and learning 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 [5]. Users of search engines have been accustomed to using short queries with keyword combinations due to the restriction of interface and inner mechanism of the search engine. [5] 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 [39].

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

—A sample of 20 SMS coded FAQ query sentences are selected from a set of queries that statistically have a greater representation from the data collected from the respondents.

—Each query was designed to retrieve the five best answers. The results are verified by experienced users using datasets applied at the beginning of our experiment and their corresponding answers. The process is repeated for every query selected.

—The retrieval efficiency can be measured using *precision* and *recall*.

Precision– is the relative number of correct constituents—FAQ queries—retrieved out of those deemed 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", i.e., the structure representing the ideal analysis which the parsing results intended [17].

$$\text{Precision} = \frac{\text{Number of relevant FAQ queries}}{\text{Number of retrieved FAQ queries}}$$

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*, are inversely related: they are used to compute the unordered list of FAQ query sets [40]. They are based on the user's relevance assessments following the retrieval process [39]. 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 [20]. 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.

VII. RESULTS—COMPARISON BETWEEN *NAIVE* AND *SMSql* ALGORITHM

Naïve retrieval is done by brute force whereby the list of queries is traversed to count the frequency of occurrences of a particular word [27]. A defect of this approach is that many documents that are non-relevant appeared most. The peak of the graph is where the most relevant of the query is fetched. But before the peak, the results to the query produced many irrelevant selections. The results shown in Figures 2 and 3 are the mean values for each SMS query for the two algorithms we are considering.

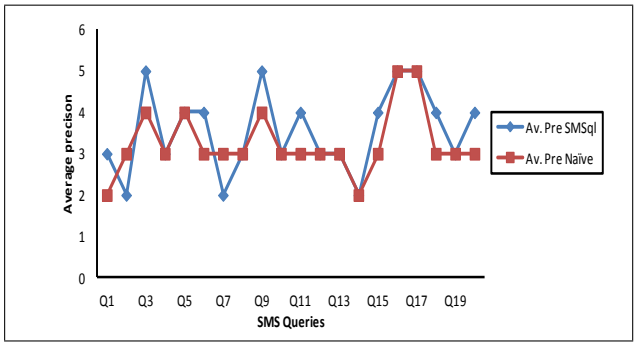

Fig. 2. Average precisions for SMS queries in the *SMSql* and *naïve* algorithms

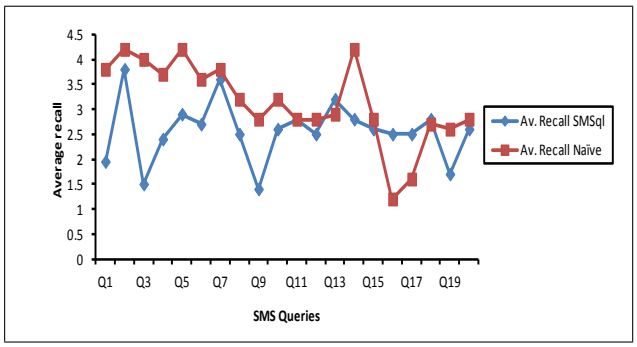

Fig. 3. Average recalls for SMS queries in the *SMSql* and *naïve* algorithms

In overall the performance of SMS queries in *SMSql* and *naïve* techniques may appear very difficult to confirm. There

is 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 two methods. This is done by comparing 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 at two different occasions, or when there are matched pairs [41]. A *paired-samples t-test* was conducted to evaluate the average precision for *SMSql* and *naïve* algorithms. There is a statistically insignificant difference in the performance of *SMSql* (Mean= 3.55, Standard Deviation= 0.9987) and *naïve* (Mean= 3.25, Standard Deviation= 0.7864); $t(19) = 2.042$, $p > 0.005$ (p-value= 0.55) at confidence interval of 95%.

VIII. CONCLUSION

We compared the retrieval efficiency of two algorithms—*SMSql* and a *naïve*—in an SMS-based FAQ system, in terms of matching terms of a specific query to its relevant answer. Statistically, there is no significant difference in the two techniques. We intend to compare *SMSql* with other retrieval algorithms, e.g. *tf-idf*, in future experiments. The system is expected to produce relevant answers to the normalized SMS query. If the searching process does not provide a relevant document for the user's information, the user can then modify and reformulate the query. This work is based solely on monolingual English, our further work will involve bi-lingual and cross-lingual FAQ systems, using other South African languages for our enquiries.

ACKNOWLEDGMENT

The authors will like to appreciate the Research Committee of the University of the Western Cape, Cape Town for funding.

REFERENCES

[1] 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.

[2] M. L. Mphahlele and K. Mashamaite, "The impact of short message service (SMS) language on language proficiency of learners and the SMS dictionaries: A challenge for educators and lexicographers," in *IADIS International Conference on Mobile Learning*, 2005.

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

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

[5] W. Dayong, Z. Yu, Z. Shiqi, and L. Ting, "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.

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

[7] M. Agoyi and D. Seral, "SMS security: An asymmetric encryption approach," in *6th International Conference on Wireless and Mobile Communications (ICWMC)*, 2010, pp. 448–452.

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

[9] D. Zurovac, A. O. Talisuna, and R. W. Snow, "Mobile phone text messaging: tool for malaria control in africa," *PLoS Medicine*, vol. 9, p. e1001176, 2012.

[10] M. Meingast, T. Roosta, and S. Sastry, "Security and privacy issues with health care information technology," in *Proceedings of the 28th IEEE EMBS Annual International Conference, New York*, 2006, pp. 5453–5458.

[11] W. A. Kaplan, "Can the ubiquitous power of mobile phones be used to improve health outcomes in developing countries," *Global Health*, vol. 2, 2006.

[12] 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*. ACM, 2008, pp. 67–74.

[13] 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: Association for Computational Linguistics, 2009, pp. 71–78.

[14] P. Pakray, S. Pal, S. Poria, S. Bandyopadhyay, and A. Gelbukh, "Smsfr: Sms-based faq retrieval system," in *Advances in Computational Intelligence*. Springer, 2013, pp. 36–45.

[15] Z. Elberrichi, A. Rahmoun, and M. A. Bentaallah, "Using wordnet for text categorization." *Int. Arab J. Inf. Technol.*, vol. 5, no. 1, pp. 16–24, 2008.

[16] G. Anderson, Y. Ayalew, P. Mokotedi, N. 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.

[17] 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.

[18] G. Anderson, S. Asare, Y. Ayalew, D. Garg, B. Gopolang, A. Masizana-Katongo, O. Mogotlhwane, D. Mpoeleng, and H. Nyongesa, "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] 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.

[20] A. Langer, R. Banga, A. Mittal, and L. V. Subramaniam, "Variant search and syntactic tree similarity based approach to retrieve matching questions for SMS queries," in *AND'10*, 2010.

[21] A. O. Adesina, K. K. Agbele, A. P. Abidoye, and N. A. Azeez, "Evaluating SMS parsing using automated testing software," *African J. of Comp & ICTs*, vol. 5, pp. 53–62, 2012.

[22] A. Masizana-Katongo, G. Anderson, and D. Mpoeleng, "Healthcare FAQ information retrieval using SMS," in *Prato CIRN-DIAC Community Informatics Conference 2010: Refereed Stream*, 2010.

[23] W. Y. Conwell, "Methods and systems for content processing," 2012.

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

[25] 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*, vol. 32. ACM, 1998, pp. 5–17.

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

[27] 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*, 2012, pp. 598–605.

[28] Ipoletse, *Ipoletse Training Manual for Call Centre Operators for the National Call Centre on HIV and AIDS*. Gaborone, Botswana: Ipoletse, 2002.

[29] A. S. Hornby, *Oxford Advanced Learner's Dictionary of Current English, 7th, Ed.* Oxford University Press, 2006.

[30] E. Dragut, F. Fang, P. Sistla, C. Yu, and W. Meng, "Stop word and related problems in web interface integration," in *VLDB '09*, 2009.

[31] L. Dolamic and J. Savoy, "When stop-word lists make the difference," *Journal of the American Society for Information Science and Technology*, vol. 61, pp. 200–203, 2010.

[32] I. A. El-Khair, "Effects of stop words elimination for Arabic information retrieval: a comparative study," *International Journal of Computing & Information Sciences*, vol. 4, 2006.

[33] J. Leveling, *On the Effect of Stopword Removal for SMS-Based FAQ Retrieval*. Springer, Berlin, 2012, vol. 7337, pp. 128–139.
[34] J. B. Marino, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. Fonollosa, and M. R. Costa-jussà, "N-gram-based machine translation," *Computational Linguistics*, vol. 32, pp. 527–549, 2006.
[35] A. M. Robertson and P. Willett, "Applications of n-grams in textual information systems," *Journal of Documentation*, vol. 54, pp. 48–67, 1998.
[36] A. Mogadala, K. Rambhoopal, and V. Varma, "Language modeling approach to retrieval for SMS and FAQ matching," in *FIRE 2011*, 2012.
[37] J. Michelizzi, "Semantic relatedness applied to all words sense disambiguation," Master's thesis, University of Minnesota, 2005.
[38] D. A. Hull, "Information retrieval using statistical classification," Ph.D. dissertation, Department of Statistics, Stanford University, 1994.
[39] S. Maleki-Dizaji, "Evolutionary learning multi-agent based infromation retrieval systems," Ph.D. dissertation, Sheffield Hallam University, 2003.
[40] M. Buckland and F. Gey, "The relationship between recall and precision," *JASIS*, vol. 45, pp. 12–19, 1994.
[41] J. Pallant, *SPSS survival manual: A step by step guide to data analysis using SPSS*. Open University Press, 2010.

Ademola O. Adesina received his Masters Degree in Computer Science from the University of Ibadan, Nigeria (2004). He is completing his doctoral programme in the Department of Computer Science, UWC. His research interests are in text processing, mobile computing, agent technology, information retrieval, web search and mobile security.