Development of a Facebook Crawler for Opinion Trend Monitoring and Analysis Purposes: Case Study of Government Service Delivery in Dwesa

S.I. Mfenyana[1], N. Moroosi[2], M. Thinyane[3], S.M. Scott[4]
Telkom Centre of Excellence of ICT for Development
Department of Computer Science
University of Fort Hare, Private Bag X1314, Alice 5700

ABSTRACT

For service providers it is important to know what the public thinks or requires. Having such knowledge can help in decision making and future planning. In nowadays there is very high usage of Social Networking Sites (SNSs) and blogs where people share their views, opinions, and thoughts. If service providers can keep track of such views, opinions or thoughts with regard to the services they provide, they can better their understanding about the public or clients' needs and improve the provision of relevant services. This paper presents a system prototype for performing trend analysis on Facebook. The proposed system crawl Facebook, indexes the data and provides user interface where end users can search and see the trend of topic of their choice. The main objective of this paper is to propose a framework that can contribute in the improvement of the way government official and normal citizens communicate in regard with service delivery in rural areas. The premise in this paper is that if the government can keep track of the citizen's opinions and thoughts with regards to service delivery, it can help improve the delivery of such services. This research and the implementation of the trend analysis tool is undertaken in the context of the Siyakhula Living Lab (SLL), an Information and Communication Technologies for development (ICTD) intervention for Dwesa marginalized community located in the Eastern Cape province of South Africa.

General Terms

Algorithms, Measurement, Design, Human Factors

Keywords

Data Retrieval, Social Networking Sites, Web Crawlers, Facebook, ICTD, SLL.

1. INTRODUCTION

Developing countries such as South Africa have adopted the use of Information and Communication Technologies (ICTs) to support rural development. This is usual termed ICT for Development (ICTD) and has seen many initiatives being undertaken towards socio-economic development of rural communities. One such initiative is the SLL which aims to improve social and economic situations in rural and marginalized areas [1]. This paper proposes the development of a system which will collect data feeds from SNSs from which opinion trend analysis about service delivery in rural and marginalized rural areas will be performed. The applicability of this system is wide-ranging however in this research the focus is on improving service delivery in rural and marginalized communities. A rural and marginalized community Dwesa located in the Eastern Cape Province of South Africa will be used as a field test site of this research project. This is because in this area an internet enabled ICT platform has already been deployed for use by the local schools and the community members through the initiatives of the SLL research group.

SNSs are used by people to connect with each other for business or personal purpose, and also build and reflect social relationships among people, who share similar interests and activities also used by many governments to inform, engage and serve citizens [2]. These free and easy to use cloud based applications provide individuals, organizations and societies an easy and cheaper way of communication [3]. SNSs' services provide content sharing through the publication of documents and links and also through messages exchanged using communication tools. These functionalities allow for the use of SNSs as a collaborative opinion mining platform. There are a lot of factors motivating for the use of SNSs for opinion trend analysis. SNSs are easy to use and provide a cheap way of communication as evidence by the growth of their usage [4]. Also most SNSs data can easily be accessed by the use of supported Application Programming Interfaces (APIs). In this research, collected data will be used for monitoring trends about service delivery in marginalized rural communities of South Africa. The following are factors that affect service delivery in these communities:

Access - long distances and poor road infrastructures which separate rural areas and developed regions of the country, leave populations in rural areas poorly integrated in their economies.

Cost - poor road infrastructures separating rural areas and developed regions of the country increases the costs of good deliveries and that discourages providers of goods and services from reaching the rural areas resulting in the situation where rural areas are left under-developed.

Quality - infrastructures (e.g. road, telecommunications, education, and health care centres) are a poor in most rural areas of South Africa [5]. This is a negative impact that usually chases away professional and skilled people who would bring development in such areas.

The SLL, wherein this research is being undertaken, is an ICTD initiative which was developed to provide ICT solutions for most of problems Dwesa community is facing through e-Commerce portals, e-Learning solutions, e-Government services and e-Health applications [5]. Dwesa is a coastal region located in the previous homeland of the Transkei in the

Eastern Cape, South Africa. Residents of Dwesa have to travel more than 40km to visit different government offices at the Willowvale Business Centre and approximately 80km to the Idutywa Business Centre to submit the complaints or access public services [6]. Taking into account the fact that rural areas of South Africa consist largely of unemployed people, the costs of reaching municipal offices become prohibitively expensive.

The proposed system will provide a cost effective solution, which will enable the local government in Dwesa (and other communities in South Africa) to determine the trending topics and discussions that are related to service delivery within their jurisdiction. This is achieved through the implementation of a focused Facebook crawler and opinion trend analysis tool. Focused crawler is a topic-driven web crawler which selectively retrieves relevant web pages to a predefined set of queries or topics [7]. The crawler extracts status and comments feeds which are then used for trend analysis purposes.

The remainder of the paper is organized as follows. The overview of the research and the background and related work is given in section 2 and 3 respectively. Section 4 presents research methodology. Section 5 presents system architecture and implementation. Section 6 experimental results and last section is the conclusion and future work.

2. RESEARCH BACKGROUND

This section is meant to give necessary background that will help readers understand the research domain of this paper which is social web mining, with application in opinion trend analysis.

2.1 Social Networking Sites (SNSs)

Two SNSs definitions from literature were chosen, one by Boyd and Ellison where they defined SNSs as networks which allow individuals to "construct a public or semi-public profile within a bounded system, articulate a list of other users with whom they share a connection, and view and traverse their list of connections and those made by others within the system" [8]. Provan et al define SNSs as networks made of a set of nodes and a set of ties. Nodes being the individuals, organizations or societies, while ties represent a particular type of relationship between the nodes such as friendship, family, and co-workers [9]. SNSs are ideal examples of Web 2.0, not only because of their social networking aspects which include the user as a first class object, but also due to their use of new user interface technologies. Web 2.0 is an umbrella term that is used to represent Web sites which incorporate a strong social component; involving user profiles, friend links or Web sites which encourage user generated content in the form of: text, video, and photo postings along with comments, tags, and ratings [10].

Web Crawlers

A web crawler is a computer program that browses the World Wide Web in a methodical, automated manner or in an orderly fashion [11]. Web crawler are used by search engines to create a copy of all the visited pages, which it saves on a single directory and subsequently indexes to provide fast searches. Web crawlers are also used for automating maintenance tasks on web sites, such as checking links or validating Hyper Text Mark-up Language (HTML) code. Also, they are used along with data extractors to collect structured knowledge in some field by processing unstructured textual data automatically [12].

2.2.1 General Crawler architecture

In this section we discuss the general activity flow in the web crawling process. This will help to give a clear overview of how a web crawler functions. In its simplest form a crawler starts with one or few Uniform Resource Locator(s) (URL) to reach specific web pages, from there it use the links from such pages to reach other pages [13]. In general every crawler consists of the following:

Frontier - the list of seed URLs or user account IDs in SNS such as Facebook, populated by a user or other program as non-visited nodes. This list is updated as the crawling process is occurring by adding newly found URLs to the to-be visited list.

Crawling loop – this is the stage where most of the crawling process occurs. When web crawling initiates, the web crawler start by selecting the next URL(s) or user account ID from the to-be visited from the frontier. It then fetches the corresponding pages through a Hypertext Transfer Protocol (HTTP) request and extracts relevant information from the web page. If new URLs or user account IDs are discovered they are added to the to-be visited list of URLs in the frontier [13].

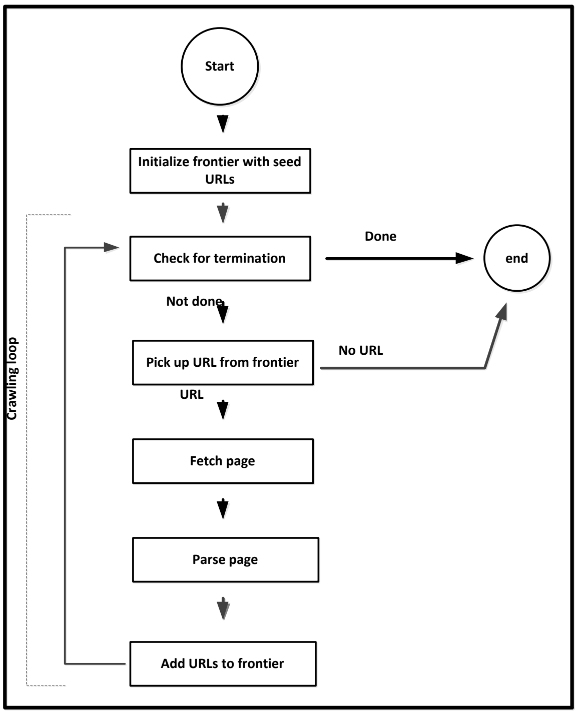


Figure 1: General Crawler Architecture [13]

Referring to **Figure 1**, the following are the processes which occur during crawling process on their preceding order:

1. Initialize frontier with seed URLs: The frontier is the to-be visited list of a crawler that contains the URLs of unvisited pages.

2. Check for termination: Checks if the frontier is empty, if not returns the next URI to be analysed.

3. Fetch page: Obtain the web page through an HTTP client request.

4. Parse page: Creates a parser on the page to extract useful information and possibly guide the next step of the crawler, and also eliminate irrelevant information.

Opinion tracking and Trend analysis

SNSs' data can be a valuable source of data for opinion trend analysis if mined and filtered properly. This data can be used for many purposes such as customer relationship management, public opinion tracking about politics, sports. Through these blogs and SNSs, subjective information is generated by people expressing their views and opinions based on various topics or subjects.

A good example of subjective information collection can be found on blogs on "Yahoo news" which amongst other topics, focuses on various aspects of political covering the entire spectrum of politics related issues [14]. In this arena normal citizens express their opinions on everyday political issues, politicians communicate their ideas, journalists criticizing the government and all this data is open to all. Such data can be gathered through systems which can automatically track opinions of the public enabling decision makers to make sense of the enormous body of opinions expressed on the SNSs and traditional blogs.

The process of tracking most emerging topics or events that attract the attention of a large fraction of Blogs or SNSs users is called trend analysis [15]. Trend analysis by definition is the analysis of changes over time. This research mainly focuses on opinion trend analysis, it proposes to develop a system which will help government officials to see most trending government service delivery related topics on Facebook.

3. RELATED WORK

In this section, a review of prior work on crawling social networking sites, web mining and sentiment analysis is given.
Most existing research papers on SNSs are based on SNSs structure analysis and sentiment analysis. This research adds a feature of opinion trend analysis. This section start by giving the background of SNSs data mining, how it started till today as it has attracted many researchers from different fields. In 2012 [16], the editors of Special Issue on a Decade of Mining the Web, provided a brief overview of how Web mining evolved from the first Web mining workshop (WEBKDD'99). The workshop took place at the KDD'99, with the theme International Conference on Discovery and Data mining. They mentioned that during that time web data mining challenges and opportunities were very different compared to ones of these days. Social Web by that time did not exist, Semantic Web was emerging. The major aim of web mining was to understand what users want and to help them to perform simple tasks inside web sites. In 2008 in the 10th WEBKDD workshop the series of core web mining were closed because they reached maturity from there the research domain of web mining shifted to topics of knowledge discovery such as recommendation engines, model adaptation for user profiling, understanding communities and monitoring their evolution, modelling and interpreting user searches [16]. In 2102**,** Boldrini et al did their research with the aim of creating EmotiBlog, a fine grained annotation scheme for labelling subjectivity data in non-traditional textual genres [17]. Their research was motivated by massive data which is available on Web 2.0, which is made available by people through the use of new communication tools such blogs, forums and reviews which people from all over the world employ as source of information in addition to traditional textual genres such as newspaper articles. They argue that it can be difficult to identify subjective data because of the mixture of text styles, a wide range of topics and sources, multiple languages, grammar and spelling mistakes, informal language, use of colloquialisms or slang, extensive amounts of data, continuous updating of information. The other part of their research was to develop technologies to organize textual information, not just in terms of topical content, but also taking into account the emotions and opinions embedded, as well as the source of the discourse. They confined their scope of research by only focusing on three languages (English, Spanish and Italian) and three topics. The main objectives of their research were to: design a fine-grained annotation schema able to capture the linguistic means of affective expression in non-traditional textual genres, annotate a collection of blog posts using the resulting schema, and evaluate the robustness of the scheme creating Machine Learning models using the annotated elements [17]. In 2011, Papadopoulos *et al* proposed a survey for community detection in the context of social media [18]. Their objective was to address two aspects which they felt are not addressed in the context of social media which are: performance aspects of community detection methods, namely computational complexity, memory requirements and possibility for incremental updates of already identified community structure, and interpretation and exploitation of community detection results by Social Media applications. They elaborated on methods for detecting and monitoring communities as the SNS evolves. They were focusing particularly on the issue of algorithm performance, but they also elaborated on different applications of community detection, such as user profiling, event detection and tag disambiguation. In 2012, Tuzhilin did a research survey on Customer relationship management (CRM) and Web mining: the next frontier [19]. The argument of the research was that CRM can advance the series of Web mining field for the next decade even though CRM in mid-2000s there was a period where it was ineffective due to various reasons, including (but not limited to) high failure rates of various CRM projects, disillusionment with CRM systems, and disappointments with the results generated by various CRM applications in the industry. In 2012, Hagberg developed a news reading system [20]. The system was responsible for delivering interesting news to specified user based from the previous activities on Facebook. The system crawl Facebook for user information and subsequently retrieves relevant news for the specified user.

4. RESEARCH METHODOLOGY

The research method that was followed in this research consists of a combination of well-established research methods which are requirements gathering, system development and system testing. Such research methods informed the decision of using Iterative Incremental development approach. This development approach allow to develop the system in different modules which can be incremented and also revised later if research objectives are not yet meet. Through literature review of similar previous works on this research domain, it was possible to design and foresee the appropriate tools which could be used to develop the proposed system. The system is developed specifically for Dwesa community but the system can be easily deployed for use in any rural area with Information and Communications Technology (ICT) access; upon that to understand the community so it can be possible to properly specify the functional and non-functional system requirements. Data collection through informal interviews was conducted, with the objectives of knowing; how people of Dwesa community are currently reporting the issues regarding government service delivery to their local government and what are the drawbacks on the system they are using. The system was then developed with open source technologies, because SLL where the system will be deployed uses open source software and because they are free. Review of programming languages and most of the tools which were previously used to develop similar system was carried out as well. The system was then developed in different modules which were constantly tested and later integrated together to form one system package. The following section describes how the method was executed and provides brief details of system implementation.

5. SYSTEM ARCHITECTURE AND IMPLEMENTATION

The developed framework serves two high level functionalities which are outlined as follows:

1. Data extraction from Facebook, text extraction, text pre-processing and text indexing.

2. Index searching which consist of two sub-modules content matching and frequency analysis sub-module and keyword search sub-module, both these are integrated together with one user interface. The user interface is developed in the manner that it eases the usage of the system.

These high level system functionalities and the undertaken literature review informed the design of the system. The following is the high level system architecture of the proposed system.

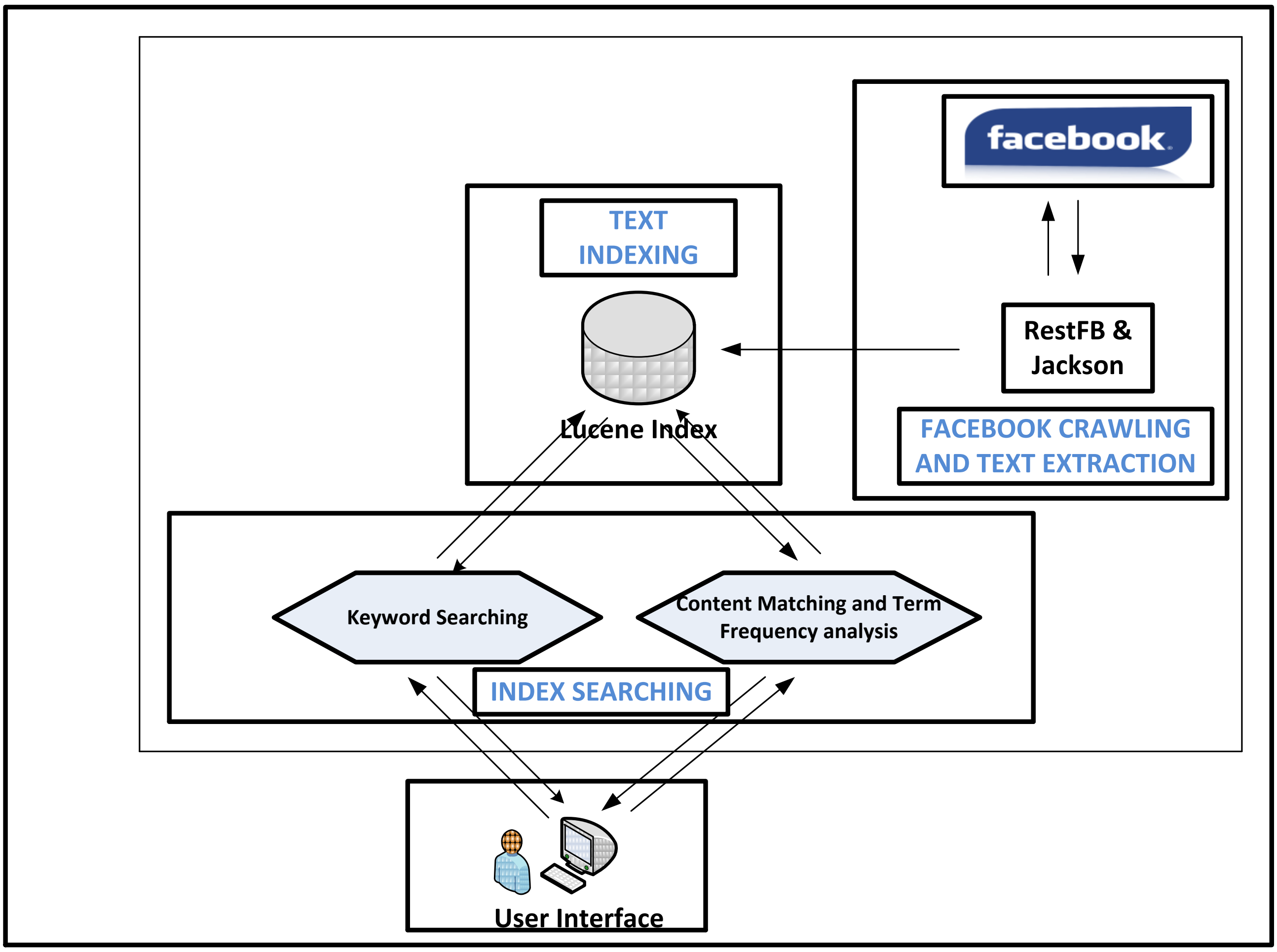


Figure 2: System Architecture

The system architecture above illustrates the internal and external structure of system modules integrated together in one package to form one system. The following subsections provide a brief background overview of the tools used and the implementation details of the different modules of the developed system starting from the back-end to the front-end. The whole system was developed on Linux Ubuntu operating system and Netbeans IDE platforms because they are open source technologies.

Facebook Crawling and Text Extraction

This is the system back-end module developed on top RestFB and Jackson java libraries. RestFB is a simple client java based library alternative for the Facebook Graph API written in Java [21]. This library is used by developers for communicating with Facebook database servers. FacebookClient interface was used to communicate with Facebook Graph API. This interface specifies how the Facebook graph API client supposes to operate [22]. The sub classes FetchConnection and FetchObject of FacebookClient were used for extracting friends' user account ID's and data feeds from Facebook respectively. All data extracted from Facebook was return in JsonObjects, an unordered collection of name/value pairs. JavaScript Object Notation (Json) is a lightweight data-interchange format between machines and humans [23].The data was also filtered by specifying the parameters to retrieve from Facebook and that eased the process of data mapping through java beans when parsing text. The extracted Facebook data was then parsed to plain text using Jackson library. The parsing of Facebook data for two reasons:

1. The user account IDs were parsed to get actual value of user account ID. The user account ID was then used as reference to retrieve the latest status update together with associated comments of the specific user.

2. The data feeds which is the actual data indexed, was parsed because Lucene only index plain text.

Lucene is an open source java library developed by the Apache organization for high text indexing and efficient search algorithms performance was used for indexing and adding searching functionality on system. Collecting data from Facebook Breath-first-search (BFS) SNS sampling algorithm was used, which includes an agent that collects seed's friends user-ID's and an agent which is responsible for crawling friends' data feeds. In BFS sampling algorithm web pages are crawled according to the way they are discovered [24].

The Facebook Crawling and Text Extraction module operates as follows: it contacts the Facebook server, providing Facebook access token required for the authentication and permissions for accessing users' data. Once logged in, the agent starts crawling seed's (logged in Facebook user) friends list extracting friends' user-IDs and also crawling data feeds of each and every user in First-In-First-Out (FIFO) queue manner. Facebook data feeds are arranged in a chronological order with the most recent data feed appearing at top, and only the top data feed that is the last status update in every friend's Facebook account is crawled and index. This allows the tracking of the trending topic, on the network of the seed node.

Text Indexing

This system module was built on top Lucene information retrieval java library. The Facebook Crawling and Text Extraction module as aforementioned was developed for Facebook data extraction and text parsing; the text was then indexed using Lucene. Lucene only indexes data available in textual format. The input text data in Lucene is stored in an inverted index data structure, which is stored on file system or in memory as a set of index files. On the development of this module we used different Lucene library classes to accomplish the objective of developing a working index. The following are the core indexing classes which were used during the development of the index: Directory, Analyzer, and IndexWriter. Directory is the abstract class used to represent the location where the index files are to be or stored. We also used its subclass FSDirectory store index files in the actual file system. This subclass was used because we were getting large sizes of data and such data was also intended to be used for trend analysis purposes. Analyzer class was used for converting the text data into a fundamental unit of searching, which is called a term. During analysis, the text data goes through multiple operations such as extracting the words, removing common words, ignoring punctuation, words stemming, also converted into tokens, and these tokens are added as terms in the Lucene index. There are already built-in Lucene analyzers which differ in the way they tokenize the text and apply filters, but in our development we used the StandardAnalyzer. IndexWriter class was used for creating and maintaining an index and its constructor accepts a Boolean that determines whether a new index must be created or an existing index is opened [25]. IndexWriter class also provides methods for adding, deleting, or updating documents in the index. To avoid duplications and ensuring that most recent statuses were used, we used DeleteAll method every time new data was about to indexed. Lastly, Document class was used to represent a collection of fields, and the data was stored under two fields "title" and "content". Title being status or Comments and the content being the actual data we obtained from Facebook.

Index Searching

This module defines how end user's queries and trend analysis is done. This module from the proposed system consists of two sub-modules keyword searching, and content matching and frequency analysis which serve the purpose of trend analysis.

5.3.1 Content Matching and Frequency Analysis

Pranav [26] classifies trend detection methods which currently being used, from a text data, as fully-automatic and semi-automatic.

Fully-automatic - these are the systems that accept the input collection of textual data and generate a list of emerging topics. These systems provide the graphical visualization which helps the end user to examine the actual emerging trends based from the evidence provided by the system [26].

Semi-automatic - these are the systems that require a user input such as a topic and then outputs the evidence that helps in determining whether that topic is emerging or not. Such systems provide a descriptive report of the available evidence [26].

The proposed system on this paper adopted fully-automatic method and term-document matrix. Term-document matrix is the frequency of terms or words in a collection of documents, documents being columns and rows representing terms. The system was provided with the collection of textual data in an array, which are related to service delivery. This sub-module then, starts by checking if the words from the array are also available from the index. Words which are found from the array and the index, their frequencies are then calculated. To access data from Lucene index MatchAll query was used to retrieve all the documents from the Lucene index. Then tokenized the retrieved data so can be possible to match terms from the array and tokenized terms from index and also be able to calculate words frequencies which are found matching.

Keyword Searching

This sub-model enables the keyword searches. Based from the results from the above described sub-module an end user can search the most trending words through this system feature, to get the content where they can see what has been said on Facebook about services delivery. Unlike, Content Matching and Frequency Analysis sub-module which restricts the end results since it uses data provided to as training data set, keyword searching accepts any query and retrieves the documents that contain those words. To limit the number of documents retrieved we used Lucene Topscoredoccollector class and limited the results to top 10 documents with high hits scores.

User Interface

The user interface was created using JavaServer Pages (JSP) and linked to the two index searching sub-modules with a java servlet. The servlet is responsible for accepting queries from JSP and parse them to index searching sub-modules and accepts the feedback from these sub-modules and parse back to be displayed on JSP page. Jfreechart bar graphs are used to visualize results from content matching and frequency analysis sub-module and results from keyword are displayed in play text. The following section discusses the initial system testing and the results.

6. SYSTEM TESTING AND RESULTS

The modules of the system were constantly tested during their development. Collecting data feeds and parsing them to plain text and indexing was successfully implemented and functionally tested on the system. Integration testing was used to test the whole integrated system.

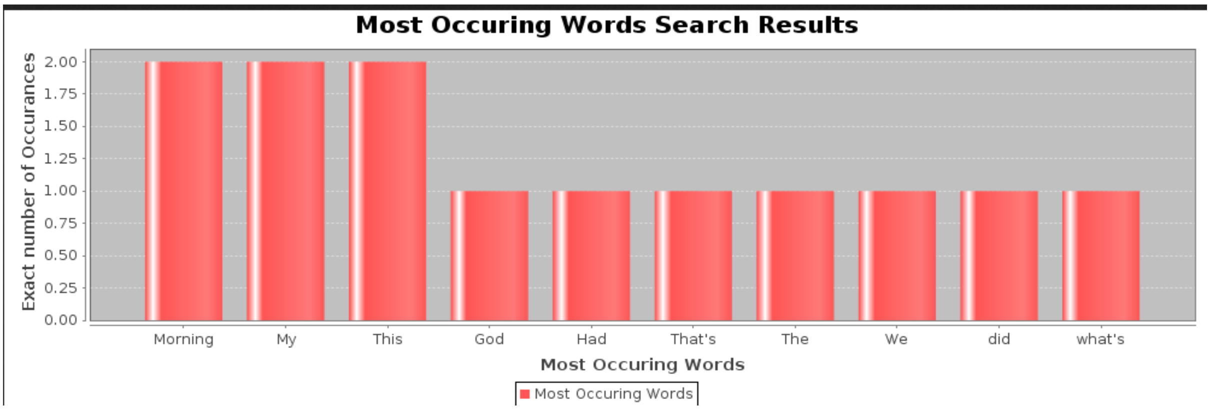
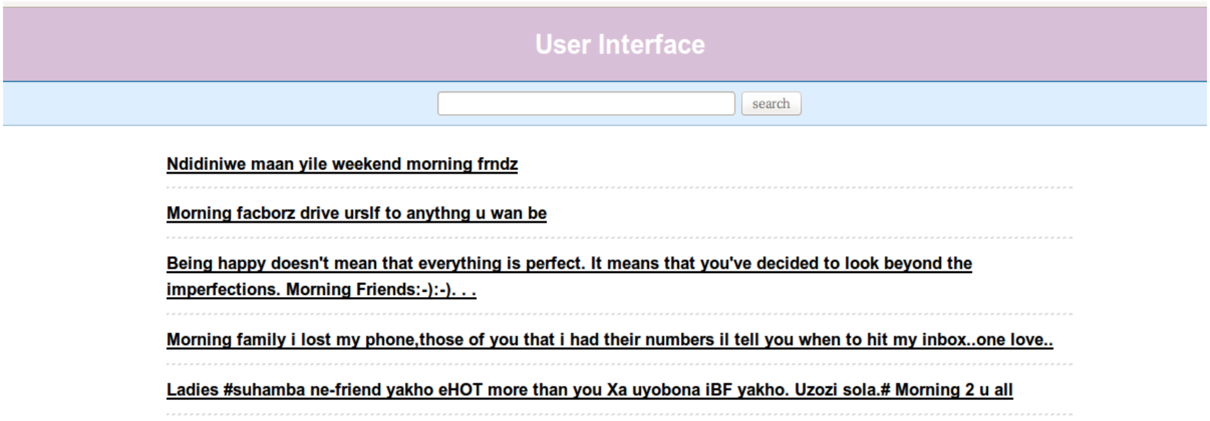


Figure 3: Word Frequency Visualization

Testing the functionality of the content matching and frequency analysis sub-module the system was supplied with this data { "bad", "good", "house", "can't", "cannot", "couldn't", "didn't", "do", "doesn't", "doing", "don't", "down", "during", "had", "having", "he", "he'd", "did", "he's", "her", "here", "here's", "hers", "herself", "him", "himself", "his", "how", "week", "I'd", "I'll", "if", "in", "into", "Birthday", "itself", "me", "more", "most", "my", "man", "That's", "Morning", "Beautiful", "God", "ought", "the", "family", "phone", "weak", "themselves", "then", "this", "we", "church", "were", "weren't", "what", "what's", "is", "when"}. Figure 3, shows the results of the words which were both found from the training data set and the index. The use of these words was specifically for system functional testing and the results were satisfactory because the system could restrict its frequency counting only to the supplied data and could visualize the data for end user to examine the emerging terms. In future words which are government services delivery related will be used, as the paper domain research entails. The results from Figure 3 could also be used as guidance for searching the most emerging topics from Facebook. The system was also tested by using one of the words from Figure 3. The word “morning” as the word with the higher frequency from the results of content matching and frequency analysis sub-module was used to test keyword searching sub-module.



Ndidiniwe maan yile weekend morning frndz

Morning facborz drive urslf to anythng u wan be

Being happy doesn't mean that everything is perfect. It means that you've decided to look beyond the imperfections. Morning Friends:-):-). . .

Morning family i lost my phone,those of you that i had their numbers il tell you when to hit my inbox..one love..

Ladies #suhamba ne-friend yakho eHOT more than you Xa uyobona iBF yakho. Uzozi sola.# Morning 2 u all

Figure 4: Keyword Search Result

The above results on Figure 4 were found after searching for the specified keyword (morning). Currently the system uses any random words not the exact words which are government service delivery related, because the system was tested on Facebook account where friends are not aware of the developed system and they hardly ever post anything related to government services. Lastly, any word can be searched not like the system is only restricted from the results found from content matching and frequency analysis sub-module.

7. CONCLUSION AND FU TURE WORK

This paper described the tool which is intended to be used for collecting, indexing and visualizing data from Facebook. This paper also discussed and presented the initial implementation of a system for extracting and indexing information from Facebook. The system allows for the querying of the extracted information to determine the opinions and comments about specific topics and terms. The tool also has a frequency analysis module and visualization component which support easy presentation of the information retrieved from the SNSs. Further extension of the tool involves collecting and supplying words related to government services delivery and further system optimization.

8. ACKNOWLEDGMENTS

This work is based on the research undertaken within the Telkom CoE in ICTD supported in part by Telkom SA, Tellabs, Saab Grintek Technologies, Easttel, Khula Holdings, THRIP and National Research Foundation of South Africa (UID : 84006). The opinions, findings and conclusions or recommendations expressed here are those of the authors and none of the above sponsors accepts any liability whatsoever in this regard.

9. REFERENCES

[1] “Project Activities | Siyakhula Living Lab.” [Online]. Available: http://siyakhulall.org/?q=activities. [Accessed: 20-Sep-2013].

[2] M. Srivastava, “Social Media and Its Use by the Government,” J. Public Adm. Gov., vol. 3, no. 2, pp. 161–172, Jul. 2013.

[3] R. Bassett, T. Chamberlain, S. Cunningham, and G. Vidmar, “Data Mining and Social Networking Sites: Protecting Business Infrastructure and Beyond,” Data Min. Soc. Netw. Sites, vol. Volume XI,, no. 1, pp. 352–357, 2010.

[4] E. Costa, R. Ferreira, P. Brito, I. I. Bittencourt, O. Holanda, and A. Machado, “Expert Systems with Applications A framework for building web mining applications in the world of blogs : A case study in product sentiment analysis,” Expert Syst. Appl., vol. 39, no. 5, pp. 4813–4834, 2012.

[5] M. Thinyane, H. Slay, A. Terzoli, and P. Clayton, “A Preliminary Investigation into the Implementation of ICTs in Marginalized Communities,” in SATNAC, 2006, p. No 213.

[6] B. T. Jakachira, “Implementing an integrated e-Government functionality for a marginalized community in the Eastern Cape , South Africa Master of Science in Computer Science University of Fort Hare by,” no. November, 2009.

[7] S. Batsakis, E. G. M. Petrakis, and E. Milios, “Improving the performance of focused web crawlers,” Data Knowl. Eng., vol. 68, no. 10, pp. 1001–1013, 2009.

[8] N. B. Ellison, “Social Network Sites: Definition, History, and Scholarship,” J. Comput. Commun.

[9] K. G. Provan, a. Fish, and J. Sydow, “Interorganizational Networks at the Network Level: A Review of the Empirical Literature on Whole Networks,” J. Manage., vol. 33, no. 3, pp. 479–516, Jun. 2007.

[10] K. Differences, G. Cormode, and B. Krishnamurthy, “Key Differences between Web1.0 and Web2.0,” pp. 1–30, 2008.

[11] S. S. Dhenakaran and K. T. Sambanthan, “Web Crawler - An Overview,” vol. 2, no. 1, pp. 265–267, 2011.

[12] B. Pontes, S. Rocha, R. Luis, and T. Santos, “Example-Driven, Content-Based, Distributed Crawling and Extraction for Online Social Network Analysis,” 2006.

[13] G. Pant, P. Srinivasan, and F. Menczer, “Crawling the Web,” springer, pp. 153–177, 2004.

[14] “Yahoo News ZA - Latest World News South Africa News Headlines.” [Online]. Available: http://za.news.yahoo.com/. [Accessed: 20-Sep-2013].

[15] M. Mathioudakis and N. Koudas, "TwitterMonitor : Trend Detection over the Twitter Stream," pp. 5–7.

[16] M. Spiliopoulou, B. Mobasher, O. Nasraoui, and O. Zaiane, "Guest editorial: special issue on a decade of mining the Web," Data Min. Knowl. Discov., vol. 24, no. 3, pp. 473–477, Mar. 2012.

[17] E. Boldrini, A. Balahur, P. Martínez-Barco, and A. Montoyo, "Using EmotiBlog to annotate and analyse subjectivity in the new textual genres," Data Min. Knowl. Discov., vol. 25, no. 3, pp. 603–634, Mar. 2012.

[18] S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos, "Community detection in Social Media," Data Min. Knowl. Discov., vol. 24, no. 3, pp. 515–554, Jun. 2011.

[19] A. Tuzhilin, "Customer relationship management and Web mining: the next frontier," Data Min. Knowl. Discov., vol. 24, no. 3, pp. 584–612, Feb. 2012.

[20] J. Hagberg, "Development of a News Reading System," 2012.

[21] "RestFB - A Lightweight Java Facebook Graph API and Old REST API Client." [Online]. Available: http://www.restfb.com/. [Accessed: 20-Sep-2013].

[22] "FacebookClient (RestFB)." [Online]. Available: http://restfb.com/javadoc/com/restfb/FacebookClient.html. [Accessed: 20-Sep-2013].

[23] "JSON." [Online]. Available: http://www.json.org/. [Accessed: 20-Sep-2013].

[24] M. Najork and J. L. Wiener, "Breadth-First Search Crawling Yields High-Quality Pages," pp. 114–118, 2001.

[25] M. Mccandless, E. Hatcher, and O. Gospodnetic, Lucene in Action, Second Edi. Manning Publications Co. 180 Broad St. Suite 1323 Stamford, CT 06901.

[26] P. Kadam, "Trend Detection and Visualization and Custom Search.".