KDSSF: A Graph Modeling Approach

Muhammad Naeem
Research Scholar (C.S)
Centre of Research in Data Engineering
(CORDE)
Mohammad Ali Jinnah University
Islamabad Pakistan

Sohail Asghar
Associate Professor
Centre of Research in Data Engineering
(CORDE)
Mohammad Ali Jinnah University
Islamabad Pakistan

ABSTRACT

In recent years, data mining applications have been found quite extendible in the area of social science like mass communication and religion studies. In traditional approach used for such work, hidden semantics between documents were not considered well. In this study, we have shown that text mining can be applied to classify social figures like politician, religious leaders. Such classification is based on text mining of speeches delivered by social figures. These social figures are famed personalities and their speeches are collected from their official websites. Our text classification is based on *tf.idf* followed by *cosine* and *Jaccard Similarity*. To improve the results on discerning features, we have designed a hash graph modeling technique Knowledge Discovery System for Social Figures (KDSSF) based on synonym words dictionary. In the comparative analysis of speeches made by social figures, we did not focus on the provision of the optimal matches but overall classification of the social figures in any domain of interests. Preliminary experiments have illustrated that inclusion of hash based graph modeling can significantly improve the results of classification.

General Terms

Text Classification, Text Mining, Machine Learning, Text Categorization

Keywords

Graph Modeling, Term Frequency, Inverse Document Frequency, Social Sciences, Synonyms

1. INTRODUCTION

Speeches made by the social celebrities including priests, political leaders and celebrities of civic society have a direct impact on tailoring the ideas, way of thinking, attitude, beliefs and nurturing doctrines of a society. Not only in matured societies but also in amateur cultures, such speeches play a pivoted role in developing or destroying a nation or group of people. These speeches are a source of inspiration as well as a hub of emanating ideologies, famous slogans and national philosophies. It goes without say that nations live in emotions and these are emotional urges which unite clusters of people to live like a nation. Inspiring speeches always boil down these emotional urges while leaving robust and vivid impact with memorable content words. With the advent of internet, an avalanche of data starts to emerge on daily basis. The digital world enables the people to archive the speeches of famous social figures while making them accessible to the public. CORPS [3] is a notable example in this regard which have made hundreds of thousands speeches available to the academics and research community. Ryder and Zhang [14] have highlighted the socio linguistic and computer collaboration with emphasis to analyze, classify and cluster such speeches. They illustrated that the research in this field is targeted towards facilitation of qualitative analysis of political communication in terms of ranking.

Document similarity has created many interests in the information retrieval community. It refers to discovery of similar documents for against a provided query. Determination of most relevant similarity between two documents is more important than accuracy and performance. Precision in such a calculation enables us to group or classify the social figures of same ideological approaches. Most notably, similarity measures between two documents enable us to infer that who is inspired by whom in chronological order. In literatures, most of the researchers have used the term frequency to find out the similarity score between two documents. Similarity measures based on term frequency give a content based matching. However, it is arguable that the usage of term frequency without considering its semantics is merely a rough document feature. Document matching based on rough document featuring provides us an approximate matching between two documents. It is investigated that the connections among terms may be overlooked resulting in dropping important semantic information of documents. Thus, there is a need of introducing a much better and effective text mining technique to give improved results based on the semantic similarity among terms found between two documents.

Graph theory, the emerging science of networks unleashes a thorough mathematically advancement to scrutinize the growth and organization of intricate systems. The usage of graph theory tools can be used to investigate how the graphical structures might influence the acquisition and retrieval of semantics of word documents. Conclusions about the role of semantically related words were drawn out through various diverse definitions of semantically relatedness. They include synonym pairs, semantic classification based on synonym, antonym or meronyms, free associates between words, core words from dictionary definitions, and co-occurring words in corpora [22], [1], [2], [13], [5].

Research in the information retrieval field has been concentrated on devising various kinds of similarity measures. In conventional approaches similarity measures are based on content of data ignoring their contextual meaning or synonyms. This results in a dire need to dig out hidden relations due to their relevant synonym effects. We have explored the relationship between any two corpuses of documents in terms of their semantics based on synonym effect. We rendered raw data from an online synonym dictionary [18] which was used for making substitution and comparison between terms of documents. Application of introduced hash graph modeling technique enabled to give an improved similarity between documents including the aspect of the hidden relationship lurking under the cover of interlinked synonyms found in documents under observations. In literature numerous functions have been discussed which described measures of similarity or distance between objects, however in this study we shall focus only two of them: the *Cosine Similarity* and *Jaccard Similarity Coefficient*. We performed our experiment on improvement of *Cosine Similarity* and *Jaccard Similarity Coefficient*.

Rest of this paper is divided into five sections. These include literature review of most relevant article followed by proposed architecture. In the result section we have elaborated result on improvements of both of the similarity measures. The last two sections are devoted to discussion followed by conclusion.

2. LITERATURE REVIEW

Most currently used techniques in literature for document retrieval and comparison are based on vector space and probabilistic models. Salton & McGill [16] introduced the vector space model which gains a serious popularity in research community. They used tf-idf scheme, in which a basic vocabulary of words or terms were used for feature extraction. Similarity between two documents was calculated on a distance function including cosine similarity [23] or Euclidean distance functions. Such vector based schemes were able to adjust the random length of term vector into a fixed length vector. Hasegawa, Kanagawa and Satoshi [8] introduced an unsupervised technique for relation discovery out of large corpora. The core idea was clustering of pairs of named entities based on the cosine similarity of the context words intervening between the named entities. They demonstrated that their method can automatically and appropriately label to the relations among named entities. However it is arguable that their technique was limited to discovery of high frequent pairs of named entities.

Gunes, Dragomir and Radev [6] proposed a stochastic graph-based method *LexRank* for computing relative importance of textual units in the domain of text summarization. Their technique investigated the concept of sentence salience to discover the central sentences in a corpus of documents. Their approach was based on computation of importance of a sentence based on the eigenvector centrality in a graph representation of all sentences within a document. The model described was having a connectivity matrix based on intra-sentence cosine similarity as the adjacency matrix of the graph representation of sentences. Latvik and Last [11] introduced a comparison between supervised and unsupervised technique for graph based keyword extraction for single document summarization. They argued that supervised classification is a better technique for keyword identification when problem space comprises of large labeled training set of summarized documents. They also showed their recommendations that unsupervised technique play its significant role when no high quality large training set is available. While performing their supervised learning based experiment by running HITS algorithm Kleinberg, [10], they used the features of the graph including *Indegree*, *Outdegree*, *Degree*, and *Frequency* of a word, distribution of words frequency, location score and tf.idf score. Ryder and Zhang [14] use the Naïve Bayesian classifier for ranking politician. The result of their ranking system identified the set of politician on the basis of various query speeches made by them. Ryder and Zhang [14] presented their ranking system with two assumptions. The first assumption was based on politician's personality. It was assumed that the politician has a vivid personal style decorated with active and powerful word of vocabulary not implicitly but also explicitly. The second assumption related to actual write-up of the speech who is usually an employee of the politician with his own negligible influence on the political speech written for his/her boss.

Motter et al., [13] illustrated that a dictionary of any language can be modeled into a graph. Upon analysis of such graph, some interesting features were revealed. These features were all related to its comparison to scale free and random networks. Author argued that linguistic network exhibit small world effect as well as a sparse graph. However the experiment was performed with exclusion of some of the outliers' data. The underlying parameters for the comparison to other networks considered by them include clustering coefficient and shortest path length. Ferrer and Ricard [5] argued that the common lexicon is the subset of whole of the lexicon of a language. They described that the common lexicon which serves as the kernel of any regular lexicon is a mandatory requirement for basic successful communication. Kernel lexicon has enabled researchers to divide lexicon into basic and specialized words.

Tommy, Zhang and Rahman [20] presented document representation by means of vectorized multiple features including term frequency and term-connection-frequency. They illustrated that a document can be expressed by undirected as well as by a directed graph. The vectorized graph representation enables the extraction of terms by applying various feature extraction techniques. They also showed that the hybrid document feature representation is better suited towards extracting underlying semantics with improved accuracy which is usually not easy to obtain out of term histograms. Michael and Vitevitch [12] graphically modeled adult lexicon using *Pajek*, a program for large network analysis and visualization [2]. They modeled words as vertices and their connection represents the phonological relatedness. They showed that such a graph exhibits small-world characteristics.

Jaccard Similarity Coefficient has been used in various domain. Hamid and Manucher [7] evaluated the performance of Jaccard's similarity coefficient with the production data-based similarity coefficient. They argued that Jaccard Similarity coefficient is its incapability of considering numerous kinds of production data in the machine component grouping process. They proposed a modified version of Jaccard's similarity coefficient namely *production data-based similarity coefficient* to surmount this problem. Jacob and Benjamin [9] worked on Wikipedia data to calculate the Jaccard similarity coefficient for pairs of users and pairs of pages. They developed a tool for analysis of the large amount of data associated with social networks communities on the web. Their tool was based on co-occurrence of page edits while generalized in a way to compute the Jaccard similarity between entities in any arbitrary column of a data set of co-occurrence with other arbitrary columns. Major task of their tool was aimed towards performance issues related to calculation of Jaccard similarity in huge volume of dataset.

Ehud and Somayajulu [4] described an empirical analysis of near-synonym choice in weather forecast system. They argued that the semantic nature of the near synonym choices are not the only choices but there are other factors responsible for playing more important role in decision making of weather forecast system. These included preferences and idiosyncrasies of individual authors; collocation; variation of lexical usage; and position of a lexeme in a text [4]. They also suggested that the word of context can be extended to various factors in a specific domain. Vincenzo, Marco and Domenico [21] described an experimental semantic approach for mining knowledge from the World Wide Web. Their objective was to extract a context-specific knowledge out of web documents. They proposed the idea of connected text words in a specific domain to develop a dictionary of indexing structure. They used WordNet [24] lexical database as reference knowledge for the English web documents and showed that the context specific knowledge information retrieval system is much more efficient than conventional information retrieval.

3. PROBLEM STATEMENT

We are given with following set of objects.

1. Given a set of real values document vectors $T = \{t_1, t_2, t_3, \ldots t_n\}$ of variable length of magnitude and same dimensionality. Two similarity scoring functions $Sim_c(x, y)$ and $Sim_j(x, y)$
2. Given a direct graph $G(V, E)$ where
3. $V = \{v_1, v_2, v_3, \ldots v_n\}$ set of vertices
4. $E = \{<v_1, v_2>, \ldots, <v_1, v_n>, \ldots <v_2, v_1>, \ldots, <v_2, v_n> \ldots\ldots <v_n, v_1>, \ldots, v_n, v_{n-1}>\}$ such that the second vertex in each pair is a definition for first node in each pair.
5. We intend to identify and include the missing scores based on synonym effects. The similarity values exhibit commutative property such that $Sim_c(x, y) = Sim_c(y, x)$ and $Sim_j(x, y) = Sim_j(y, x)$. We also assume that magnitude of vector documents is always positive value.

4. PROPOSED FRAMEWORK: KDSSF

Bases on the literature review, we argued that conventional tf.idf, cosine similarity and jaccard similarity based measurement give similarity based on content while ignoring the context of the words.

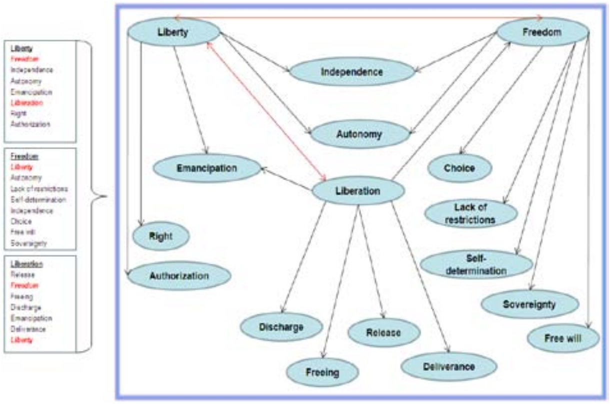

Fig 1: Graph Modeling of Word Terms from 3 documents.

In information retrieval and text mining, significant research has been performed on searching for similar words in corpora. Our approach is based on the assumption that similar words based graph modeling is to be carried out before building the document vector. This helps in improving the similarity score result as the scalar value of the vector document is enhanced by incorporating the synonym contexts. In our technique each term is treated as a node. The relationship between nodes is based on their synonymic effect. If a term *B* is in the list of synonyms for term *A* then a directed link from term *A* to *B* will exist. Once this modeling is performed. The vectors obtained as a result of tf.idf is updated in context of this graph. We explain it with an example as depicted in the figure – 1. The word *Liberty*, *Freedom* and *Liberation* all have same context. If we examine in word synonym list then they will all illustrate each other as their synonyms. However in conventional tf.idf, all these would be treated as different words rather a single concept. The hash graph based synonym similarity has the capability to incorporate each of them in their true context. This will result that the same document will treat these three words as the same word and it will increase the scalar value of document vector with occurrences of these three different but semantically same words.

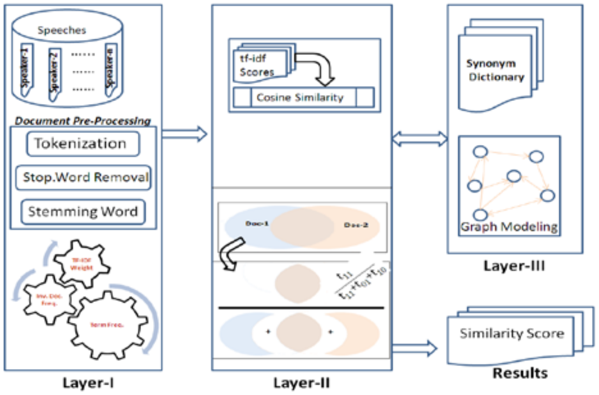

Fig 2: KDSSF: Proposed Framework for Graph Based Classification of Social Figures.

Figure-2 shows the layered approach of proposed framework. We can split up this framework into following steps:

4.1 Preprocessing

This step is primarily a preprocessing step in which corpora of various speeches are collected in plain text. Like most of the data mining task, preprocessing is usually sharing much of time and same is true in our case. This step involves necessary action of tokenization, removal of stop words and performing stemming word. This step is divided into following three sub steps:

4.1.1 Stop Words

Stop words are irrelevant terms in context of prime subject of the text document regardless of how frequently they are found in the document. They disseminate negligible discriminative information. Stop words include conjunctions, determiners, and prepositions.

A conjunction is described as a part of speech to join two phrases, words or clauses together. There are three types of conjunctions. Coordinating conjunctions (for, and, nor, but, or, yet, so), correlative conjunctions (both … and, either … or, not (only) … but (… also)), and subordinating conjunctions (after, although, if, unless, because).

A determiner can be defined as non lexical element preceding a noun in a noun phrase. It includes articles (a, an, the); demonstratives (this, that, these, those); possessive determiners (her, his, its, my, our, their, your); and quantifiers (all, few, many, several, some, every).

A preposition is a function word which links noun, pronoun or noun phrase to build a prepositional phrase in such a way that it can have an adverbial or adjectival relation to other word. Some prepositions are: on, beneath, over, of, during, beside.

4.1.2 Stemming Words

Stemming words can be described as those words appearing in text documents usually with different morphological variants. This results into the formulation such that if a word is not a stop word then it may be reducible to its corresponding stem word or term. The process of stemming is to acquire their *root form* while eliminating common prefixes and suffixes. The process of stemming helps us in categorizing collections of related word terms. Terms in the same cluster or group are syntactical alternation of each other while a single word can represent whole group. An example of such words are *category, categorization, categorized* and *categorizing* all share a common stem term category, thus they all can be assumed as different occurrences of one word or term.

4.2 Document Normalization

In literature, supervised learning techniques have been proposed which require largely annotated corpora for information retrieval. A serious effort is usually placed in implementation of annotation of documents. The underlying motivation behind this is that corpora usually become more efficient during classification application though at the cost increased effort of preparation with annotations. Documents are represented in proper data structures according to the arbitrary length of the document size. The important part of this step is implementation of tf.idf algorithm. This step will generate complete vector of each document. The vector of each document is scaled down to a fixed length of scalar value. The vector is composed of two values, the term frequency and the inverse document frequency

4.3 TF.IDF

tf.idf is a dot product of term frequency and inverse document frequency. To explain it formally we let D= { $d_1, d_2, d_3,.....d_n$ } such that $di \in D$. It is assumed that these documents are documents of our interest. Term frequency is the number of times a word (term) is appeared in a document. It can be described as tf_{ij} $(d_i, count(w_j))$. Document Frequency represents the number of documents having word term. It can be formally defined as: $df(w) = \sum_{i=0}^{k} d_i \, , \therefore \, w_j \in d_i$ where k is the number of all documents. j is the number of word terms found in all of the documents. Inverse Document Frequency is computed from the number of documents and Document Frequency. It can be formally defined as: $IDF(w) = \log(|D| / DF(w))$. the final value of Term Frequency Inverse Document Frequency is calculated by dot product of Term Frequency and Inverse Document Frequency. There are many search engines based on tf.idf theory.

4.4 Cosine Similarity

Cosine similarity is a popular and widely used technique in comparing documents in information retrieval out of text documents. Restricted to unit length input vectors *Cosine Similarity* which is a measure of similarity between two vectors? The parameter for this measurement is the angle between both of the vectors. In mathematics the result of the cosine similarity is from -1 to 1 where -1 indicates dissimilarity and opposition of two vectors to each other for their direction. However in text mining, value of -1 is meaningless where the return value of the cosine similarity for documents always ranges from 0 to 1. When two documents are exactly same then the value returned is 1 showing that both vectors are exactly overlapping each other at x and y coordinates. A value of 0 indicates that the angle between both of the vectors is of 90º, an indication that both vectors are exactly separated to each other. Cosine Similarity in this way also provides a means for measuring cohesion within groups or clusters of objects in text mining. Mathematically Cosine of two vectors is derived by dot product of Euclidean formula. Cosine Similarity can be formally defined as below:

$$Cos(A, B) = \frac{A \bullet B}{\| A \| \times \| B \|} \tag{1}$$

$$A \bullet B = \sum_{i=1}^{n} A_i \times B_i \tag{2}$$

$$\| A \| = \sqrt{\sum_{i=1}^{n} (A_i)^2} \quad \text{and} \quad \| B \| = \sqrt{\sum_{i=1}^{n} (B_i)^2} \tag{3}$$

In the above equations A and B denotes two document vectors. Here the document vectorization refers to the term frequency of these documents. Sahami and Heilman [15] as well as Spertus, Sahami and Buyukkokten [17] have reported that across various domains cosine similarity had produced high quality results.

4.5 Jaccard Similarity

Given two text documents *A* and *B* with identified terms, the *Jaccard* similarity coefficient measure the overlapping share between both of the documents. In each of the document either the term is found or not. The total number of possible combinations for these terms in both documents can be mathematically expressed as:

D_{11} represents the total number of terms observed in both documents.

D_{01} represents the total number of terms observed in document B only.

D_{10} represents the total number of terms observed in document A only.

D_{00} represents the total number of terms not found in either of the documents.

In text document objects, the attribute D00 has no sense as considering those terms which are not found in any of two documents is infinite and cannot be considered. Based on this fact, the *Jaccard* similarity coefficient for text document on asymmetric binary terms can be expressed as:

$$Jaccard = \frac{D_{11}}{D_{10} + D_{11} + D_{01}} \tag{4}$$

4.6 Graph Modeling

The key idea in this step is revolving around graph modeling of these terms. Graph can be considered a robust way to represent structured knowledge. Data representation in graphs can be termed as general data structure. String and trees can be considered as an instance of graph representation. Research involving learning from graphs in our study was modeled such that all of the terms are treated as nodes <i, j> and the links among these nodes are determined on the basis of their synonymic context if the second term j appears in the definition list of first term i. Such a modeling will generate a single graph. Apart from the benefit of graph modeling for improvement in document vector, this model can also be used to determine the kernel of the graph. Kernel of the graph will comprise of those terms which have been used most frequently. However for computation of kernel of graph, we need to assign label to each node where the label represents the frequency of each term. Computation of the kernel can give a common motif of a document. In our approach, directed graph is a strong and straightforward candidate to express the semantics using terms in the document. A directed graph $\vec{G}$ for a document can be formally defined as: $\vec{G} = (\vec{V}, \vec{E})$ where $\vec{V}$ denotes set of nodes which are word terms in our case. $\vec{E}$ is a set of links between word terms. This is also depicted in figure -1.

4.7 Mathematical Formalization

The last step is related to the application of similarity score. Our technique is aimed towards improvement for the input vector objects. The strength of vector objects primarily lies in their scalar values which consist of term frequency and inverse document frequency. This feature of the vector was improved in previous step using graph modeling technique. Formally we can define the application of hash based graph modeling such as:

$$HashGraph = \underset{i=1}{\overset{n}{G}}(H) \qquad (5)$$

$$Matrix\ of\ Terms = \underset{0\ \ 0}{\overset{j\ \ k}{M}} = \{t_{jk}\} \qquad (6)$$

$$Vectors\ of\ Terms = V_0^k = \underset{i=1}{\overset{n}{G}} H\ [\underset{0\ \ 0}{\overset{j\ \ k}{M}}] \qquad (7)$$

$$graphCosSim = CS_{i=0}^{k}(V_i, V_{i+1}) \qquad (8)$$

$$graphJaccardSim = JS_{0,0}^{j,k}$$

$$= \frac{\sum(t_{jk} = 11)}{\sum(t_{jk} = 01) + \sum(t_{jk} = 11) + \sum(t_{jk} = 10)} \qquad (9)$$

Where:

n = total number of nodes in hash graph.
j = Total number of terms frequency observed in all documents.
k = Total number of documents in corpus of speeches.
i = iterator.
t = terms found in the document.

It can be observed from the above equation that we have employed hash graph because it is far much efficient than conventional graph. In conventional graph each terms will require to scan through whole of the graph. However as the computational complexity of hash buckets are only 1, this result in improving the efficiency while eliminating the exhaustive search associated in graphs.

In the next section we shall explain its working in more detail describing the experimental result of proposed framework.

5. RESULTS & DISCUSSION

We have prototyped proposed solution in Microsoft C#. The logic of the proposed architecture as depicted in figure 2 is to simulate growth in similarity measure based on hash graphical modeling for synonymy effects of word dictionary. The algorithm of the architecture has a modular aspect due to its ability to accommodate graph modeling strategy. For the experimentation purpose, we picked out the regional leaders of the last 64 years of Pakistan who are believed to carry out a peculiar prevailing school of thoughts and idiosyncrasy.

Table 1: Detail of Dataset used in Proposed Architecture

Politician / Speaker		**Speech count**
Mohammad Ali Jinnah	Founder of Pakistan	55
Liaquat Ali Khan	Prime Minister	3
Zulfiqar Ali Bhutto	President & Prime Minster	24
Mohammad Zia ul Haq	President	5
Benazir Bhutto	Prime Minister	91
Pervaiz Musharaf	President	3
Asif Ali Zardar	President	3
Syed Yousaf Raza Gillani	Prime Minister	2

Corpus of speeches used in this study were collected as part of an effort to examine the effects of synonym based graphical modeling, The data was gathered from Pakistan's famous former and current political leader's web site and blogs. The big chunk of this dataset was collected from (www.ppp.org.pk, 2011; http://ismaili.net/drupal5/node/27591, 2011; http://m-a-jinnah.blogspot.com/p/speeches.html, 2011). The detail of the dataset, we chose is shown in table 1. Among these seven leaders bulk of data was available from currently ruling party's web site for Ms. Benazir and Zulfiqar Ali Bhutto. A total 11,263 comparisons were made out of this dataset in such a way that speech of every politician was matched with speeches of all other politician. The dataset was tested on Cosine Similarity, g-Cosine Similarity (Graphical Modeling based similarity), Jaccard Similarity, g-Jaccard Similarity (Graph modeling based similarity).

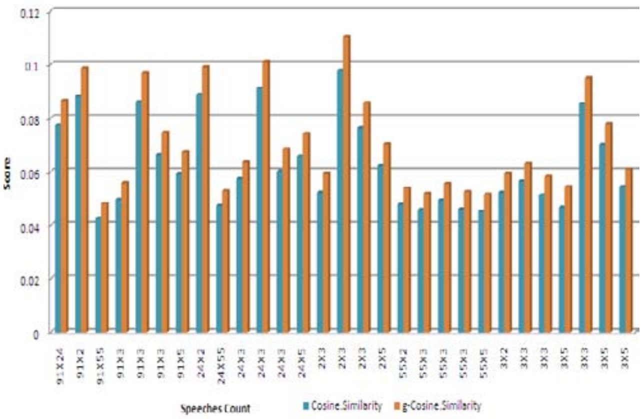

Fig 3: Comparison between Jaccard Similarity and Graph based Jaccard Similarity.

The synonym dictionary was collected from the website http://www.synonym.ca [18]. The website provides almost thirty thousand words along with their related synonyms. While building the hash graphs, stop words, stemming words and long phrases were ignored. Secondly multiwords were also simply neglected for their inclusion into the graph, since there was no uncomplicated way to treat them. This results in modeling of dictionary with almost twenty five thousand words. We have shown all of the experiments as two fold set in the figure 3 and 4. The first number in each experiment shows the number of speeches delivered by the first speaker and the second number represents the count of speeches for the second speaker. This leads us to make comparisons of total 45,052 in four experiments. From Figure-3 and Figure-4, it is evident that hash based graph modeling technique significantly improves the similarity. The improved similarity is in fact unveiling of concealed synonym effect which was the objective of this study.

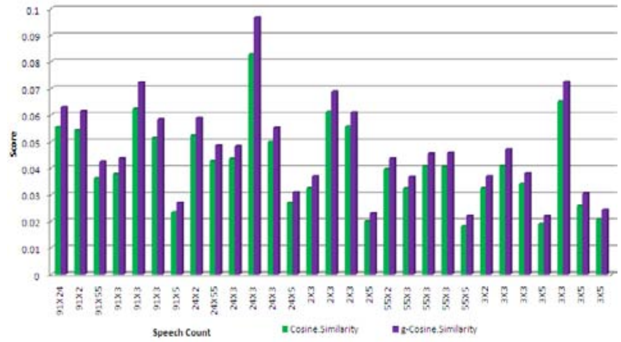

Fig 4: Comparison between Cosine Similarity and Graph based Cosine Similarity.

Another dimension and examination of the experiment is related to measuring the improvement in the similarity achieved due to inclusion of graph modeling. Figure 5 presents a comparison between both of the similarity measures. It can be concluded that in most cases cosine similarity measure got more improvement in comparison to Jaccard similarity measure with a few exceptions. We previously mentioned that there were more than forty five thousand comparisons performed. For simplicity we consolidated all of them into twenty nine comparisons for each technique such that every consolidated comparison was made between any two speakers. Analysis shows that in every set of consolidated results, percentage increase for Cosine similarity was from 9.2% to 20.8% and 10.7% to 16.6% for Jaccard Similarity respectively. This indicates that there was much more room for improvement in cosine similarity approach. The reason behind this fact is that cosine similarity by virtue of its design does not take account of the magnitude of the vector object of documents.

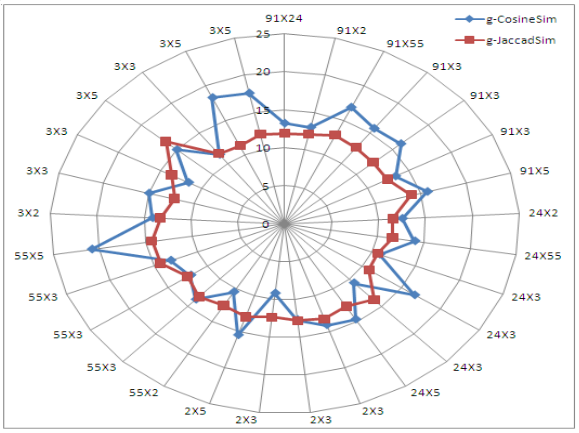

Fig 5: Improvement Comparison between g.Cosine Similarity and g.Jaccard Similarity.

6. CONCLUSION & FUTURE WORK

Intuitive Interpretation of the algorithmic technique presented in this architecture is helpful to give insight into making comparisons between any two documents. The technique can be used to develop a matrix of similarity among collection of documents of various public figures of all domains. Such matrix will eventually imply in developing clusters or classification of public figures based on their notions of ideas and preaching. The information retrieval from speeches of famous political figures can be conceived as word graphs while eliminating their superficial spelling differences and incorporating the synonym information. Such assumption has direct impact on exploring and identification of similar knowledge patterns. In order to identify context-specific elements in knowledge graphs, we introduced concept of synonym based graph mapping over all of the terms found in corpus of speeches. We have presented a preliminary assessment of the efficacy of the proposed approach.

Future work is related to considering improving the base lexicon. The dictionary used in this technique was limited to only twenty five thousand words. The quality of the synonym was also not very appreciable so there is a provision for considering using the *Wordnet* hypernymy and synonymy information for recognition of similarities among the words of document. The next stage of our methodology is related to the elimination of vagueness of the notion of *similar word* in the corpora of documents, WWW and monolingual lexicons. This notion may or may not include the impression of synonyms, near-synonyms, antonyms or hyponyms reckoning the context. The future direction is to address this problem while identifying more linguistic, and rhetorical stylistic devices including analogies, metaphors, similes, oppositions, beginning or ending rhyme, parallelism and antithesis.

7. REFERENCES

[1] Barabási, AL. Linked: The new science of networks. Cambridge, MA: Perseus; 2002.

[2] Batagelj, V.; Mrvar, A.; Zaveršnik, M. Network analysis of texts. In: Tomă, E.; Gros, J., editors. Proceedings of the 5th International Multi-Conference Information Society—Language Technologies. Ljubljana: Slovenia: Multi-Conference Information Society; 2002. p. 143-148.

[3] Corps: A corpus of tagged political speeches. http://hlt.fbk.eu/corps.

[4] Ehud R and Somayajulu S,(2004) Contextual Influences on Near-Synonym Choice, INLG 2004, LNAI 3123, pp. 161–170,

[5] Ferrer i Cancho R, Solé RV. The small world of human language. *Proceedings of the Royal Society of London B: Biological Sciences.* 2001;268:2261–2266.

[6] Gunes Erkan, Dragomir R. Radev, LexRank: Graph-based Lexical Centrality as Salience in Text Summarization, Journal of Arti_cial Intelligence Research 22 (2004) 457-479

[7] Hamid S and Manucher D, (1991) The production Data-based similarity coefficient versus Jaccard's similarity coefficient, *Computers ind. Engng* Vol. 21, Nos 1-4, pp. 263-266,

[8] Hasegawa T., Kanagawa Y. and Satoshi S., Discovering relations among named entities from large corpora,ACL '04 Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, 2004.

[9] Jacob B ,Benjamin C, (2008) Calculating the Jaccard Similarity Coe_cient with Map Reduce for Entity Pairs in Wikipedia, Wikipedia Similarity Team Project

[10] Kleinberg, J.M. 1999. Authoritative sources in a hyperlinked environment. Journal of the ACM, 46(5):604-632.

[11] Litvak M , Last M, Graph-Based Keyword Extraction for Single-Document Summarization, Proceedings of the workshop on Multi-source Multilingual Information Extraction and Summarization, pages 17–24 Manchester, August 2008.

[12] Michael S. Vitevitch, What Can Graph Theory Tell Us About Word Learning and Lexical Retrieval?, Speech Lang Hear Res. 2008 April ; 51(2): 408–422. doi:10.1044/1092-4388(2008/030).

[13] Motter A. E., de Moura A. P. S., Y.-C. Lai, and P. Dasgupta. Topology of the conceptual network of language. Physical Review E, 65(6):065102, 2002.

[14] Ryder, J., Zhang, S. (2010). Preliminary Results of Ranking Political Figures Using Naive Bayes Text Classification. Proceedings of the 2010 International Conference on Data Mining (DMIN 2010). Las Vegas, Nevada, USA. July 12-15, 2010. CSREA Press 2010. ISBN: 1-60132-138-4, Robert Stahlbock and Sven Crone (Eds.)

[15] Sahami M. & Heilman T. (2006). A Web-based Kernel Function for Measuring the Similarity of Short Text Snippets. In Proc. of the 15th Int'l Conf. on the World Wide Web, 377-386.

[16] Salton, G., & McGill, M. (Eds.). (1983). Introduction to modern information retrieval. McGraw-Hill.

[17] Spertus E., Sahami M., & O. Buyukkokten (2005). Evaluating Similarity Measures: A Large Scale Study in the Orkut Social Network. In Proc. of the 11th ACM-SIGKDD Int'l Conf. on Knowledge Discovery in Data Mining, 678-684

[18] Synonym Dictionary, http://www.synonym.ca retrieved on September, 2011.

[19] Takaaki Hasegawa, Satoshi Sekine and Ralph Grishman, Discovering Relations among Named Entities from Large Corpora, Proc. of ACL-2004 (2004), pp. 415-422.

[20] Tommy W.S., Chow, Haijun Zhang, Rahman M.K.M., A new document representation using term frequency and vectorized graph connectionists with application to document retrieval, Expert Systems with Applications, 2009

[21] Vincenzo Di Lecce, Marco Calabrese, and Domenico Soldo, (2008) Mining Context-Specific Web Knowledge: An Experimental Dictionary-Based Approach, ICIC 2008, LNAI 5227, pp. 896–905, 2008.

[22] Wilks C, Meara P, Wolter B. (2005) A further note on simulating word association behaviour in a second language. Second Language Research ;21:359–372.

[23] Zobel, J., & Moffat, A. (1998). Exploring the similarity space. ACM SIGIR Forum, 32(1), 18–34.

[24] C. Fellbaum, WordNet: An Electronic Lexical Database .MIT Press, 1998.