ISSN: 2321-7782 (Online)
Volume 3, Issue 5, May 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Automatic text summarization with context based keyword extraction

Bhavana Lanjewar
Department of Computer Engineering
G.H. Raisoni Institute of Engineering & Technology
Pune, India

Abstract: Text summarization is an important area in Natural Language Processing (NLP). It uses techniques such as keyword extraction to identify meaningful keywords from the documents. Keyword and feature extraction is a fundamental problem in text data mining and document processing for a variety of applications such as website comprehension, mobile message summarization, email categorization, etc. A majority of document processing applications directly depend on the quality and relevance of keyword extraction algorithms. The approach is based on a theory of social networks and ideas from image processing and especially on the Helmholtz Principle from the Gestalt Theory of human perception. In this paper, we talk about the implementation of Text summarization technique which uses keywords as the basic document building block and performs analysis on the document. We introduce an input to represent the context, which is used to influence the meaningfulness of the keywords. These keywords help in the automatic summarization of documents that are relevant to the given context and other documents can be termed irrelevant. We also review the outcome of experimental execution of the system to analyze the effectiveness of the keyword extraction algorithm.

Keywords: Helmholtz Principle; keyword extraction; small world topology; text Summarization.

I. INTRODUCTION

Keyword and feature extraction is a fundamental problem in text data mining and also document processing. Majority of document processing applications directly depend on the speed and quality of keyword extraction algorithms. Assignment of high quality keywords manually is expensive and time-consuming. There are various algorithms for automatic keywords extraction that have been recently proposed. Since there is no precise scientific definition of the meaning of a document, different algorithms produce different outputs. Information extraction (IE) is the task of automatically extracting structured information from unstructured and/or semi-structured machine readable documents. A broad goal is to allow computation to be done on the previously processed unstructured data which is used to summarize the text. In this paper, we talk about Text summarization as a technique which uses keywords as the basic document building block and performs analysis on the document.

Text summarization is an important area in Natural Language Processing (NLP).The manual summarization of large documents is a very difficult and time-consuming task; hence there is high demand for fast, effective and reliable automatic text summarization tools and models. This becomes especially important with an exponential growth in the number of electronically available documents on the Internet and enterprise intranets.

There are two types of summarizations, an abstractive summarization and an extractive summarization. In the abstractive summarization the goal is to represent main concepts and ideas of a document by paraphrasing of the source document in clear natural language. Automatic abstractive summarization is a very difficult task and is still in its infancy. In extractive summarization, the aim is to extract the most meaningful parts of documents like sentences, paragraphs, etc. to represent main concepts of the document. Automatic extractive summarization is a much more developed area with many different approaches and tools. In this paper, we address only extractive summarization, with the main goal being the extraction of meaningful sentences or paragraphs from the set of text documents.

The Helmholtz technique allows fast processing of large amounts of data, and thus can be easily adapted to any type of data such as speeches, emails or messages. This method is both computationally very cheap and language independent. Weighted graphs help in improving the quality of summary. We would like to evaluate the performance and accuracy of such weighted graphs for a variety of data sets. Finally, increasing the target number of edges in the graph is also important to improve the results and can be dealt with in the future.

II. Literature Survey

HITS algorithm [7] is an iterative algorithm that was designed for ranking Web pages according to their degree of “authority”. Authorities are pages with a large number of incoming links and hubs are pages with a large number of outgoing links. The HITS algorithm makes a distinction between “authorities” and “hubs”. There exist two sets of scores for each vertex, – an “authority” score, and a “hub” score.

Google’s PageRank [11] is one of the most popular ranking algorithms. It was designed as a method for Web link analysis. It combines the impact of both incoming and outgoing links into one single model. Each vertex is assigned a score identified from the execution of the algorithm, which represents the “importance” or “power” of that vertex in the graph. It is noticed that the final values are not affected by the choice of the initial value; only the number of iterations to convergence may differ. PageRank is an excellent way to prioritize the results of web keyword searches.

TextRank [9] starts by building a graph that represents the text, and connects words or other text entities with meaningful relations. For ranking entire sentences, a vertex is added to the graph for each sentence in the text. To establish connection between sentences, an edge is established for any two such sentences that share common content. A sentence that addresses certain concepts in a text, gives the reader a “recommendation” to refer to other sentences in the text that address the same concepts, and thus a link can be drawn between any two such sentences that share common content. It is based on terms co-occurrences, Parts-Of-Speech (POS) tagging is an important part of the algorithm; because without filtering, stop words are considered as keywords. POS taggers are trained, they are not language independent and POS tagging can be computationally expensive.

The implementation of LexRank [3] started with threshold on unweighted graphs. LexRank is used to define sentence salience based on graph-based centrality scoring of sentences. This summarization approach is to assess the centrality of each sentence in a cluster and extract the most important ones to include in the summary. We look into different ways of defining the lexical centrality principle in multi-document summarization, which assesses centrality in terms of lexical properties of the sentences. To determine the similarity between two sentences, it uses the cosine similarity metric that is based on word overlap and IDF weighting.

Samer Hassan [12] introduced a system that models the weighting problem as a ‘random-walk’. The new measure of term weighting integrates both the locality of a term and its relation to the surrounding context. This local contribution using a co-occurrence relation in which terms that co-occur in a certain context are likely to share between them some of their importance. In this model the relation between a given term and its context is not linear. A given term relates to a context, and the context, further relates to a collection of terms. In order to model this recursive relation, we use a graph-based ranking algorithm, namely the PageRank algorithm [11], and its TextRank adaptation to text processing [9].

Thomas Bohne [13] discusses a novel keyword extraction approach that is language-independent and self-sufficient and relies on the Helmholtz approach. The algorithm allows the analysis of documents as a whole, as well as only portion of it. Hearst briefly presents tag clouds [8] as a common visualization technique for document summarization via keywords. It not only extracts keywords but weighs them within the set of keywords with respect to a corresponding probability model. It uses tag clouds to visualize this weighting. It introduces self-regulating windows to achieve more meaningful results.

Considering the relationship between sentences and words; Xun Wang [14] and Zha [6] proposes a method for keyword extraction and summary generation by exploiting the sentence-word relationships which indicate the impact of sentences on words. Wang et al. [15] improves this method by employing three kinds of relationships: sentence-sentence, word-word, and sentence-word. The interaction between sentences and words is taken into consideration for summarization and keywords selection.

To build a concept-based graph which represents the semantic relations between sentences or words. A word/phrase can be usually linked to multiple Wikipedia concepts, so the context where it occurs is analyzed and identifies which concept it belongs to. Take the word *apple* for example, there are two concepts associated with it: a fruit name and a company name. Given there is another concept *iPhone* around *apple*, we think it is likely to be a company name. The relatedness among concepts can be measured according to their shared link-ins message [15]. If a sentence can be expressed by a concept vector, the relatedness of sentences is preserved.

Sentence similarity [10] is measured based on terms co-occurrence. In existing clustering algorithms, documents are exhibited using the vector space model (VSM). Each document is represented using these words as a vector in m-dimensional space, where m is the number of words. A big challenge to the performance of clustering algorithm is the high dimensionality of feature space. The overall sentence similarity is defined as a combination of semantic similarity and word order similarity.

Recently, a novel approach to unusual behavior detection and automatic summarization was developed in [1], [2], [4] & [5]. This approach is based on theory of social networks and ideas from image processing and especially on the Helmholtz Principle from the Gestalt Theory of human perception. First, a rapid change detection algorithm from [1] is applied to data streams and documents, based on ideas from image processing and especially on the Helmholtz Principle from the Gestalt Theory of human perception. Applied to the context of keywords extraction, it provides fast and effective tools to identify meaningful words using parameter-free methods.

III. Implementation Details

A. Problem Statement

World is observing an explosion of unstructured data today, a significant part of it, is in text form. One of the most effective text analytics techniques is extraction of keywords. Keywords provide information to obtain the summary of a document and they help in correlating documents. The main challenge is in identifying relevant keywords from the document that are of the interest. We aim to find meaningful Keywords in the desired context and create an automatic summary of documents.

The proposed system introduces a user defined input which defines the context of information being extracted, and influences the keyword extraction process. This seed is added to the set of meaningful words, which helps in automatic summarization of documents for the related context. This way, the other documents which are not related to the context will automatically be discarded, thus leading to more relevant documents.

B. System Architecture

The architecture for the proposed system is shown in the figure below. It is implemented using Python 2.7 and is evaluated on Windows platform. Python is an interpreted language and is chosen for its strong string manipulation utilities.

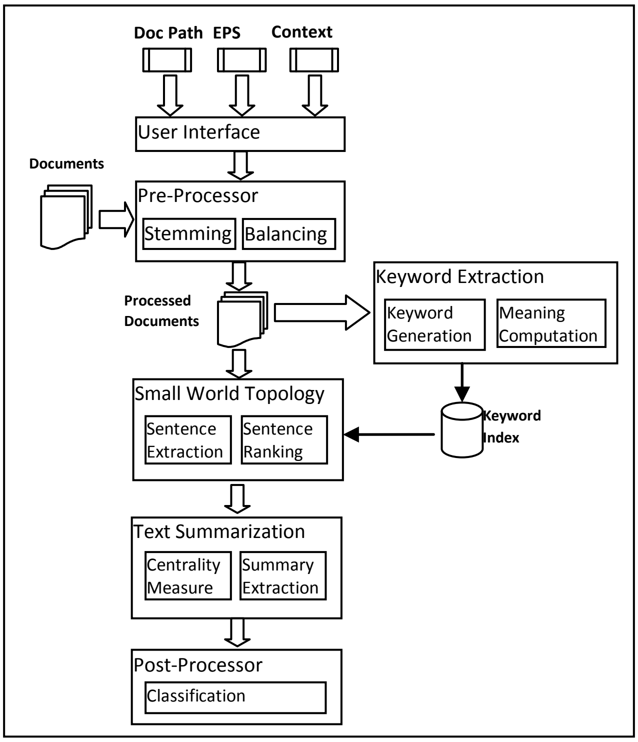


Fig. 1. System Architecture

C. System Workflow

1. User Interface: The system takes input parameters through the command line interface. Input set is represented as:

S → {D, Ɛ, C, N}

D→ $\{D_1, D_2 \dots D_M\}$ (A set of M Documents)

Ɛ → Factor to control the number of meaningful keywords

C→ Context for meaningful keywords

N→ Number of parts in the pre-processed document set

2. **Pre-processor:** The input Document set D goes through a pre-processing stage before it is fed to the keyword extraction algorithm. This is important particularly because the size of the documents can significantly impact the result. For optimal result, the documents may be divided into subsets of documents or parts such that each subset contains documents of equal no. of words. This is termed as document balancing. The output of the pre-processing is a set P containing the pre-processed documents from input document set D. Off the various schemes that can be used for document balancing; the proposed algorithm efficiently divides documents into equal sized parts of documents. The steps are shown in the flow chart below.

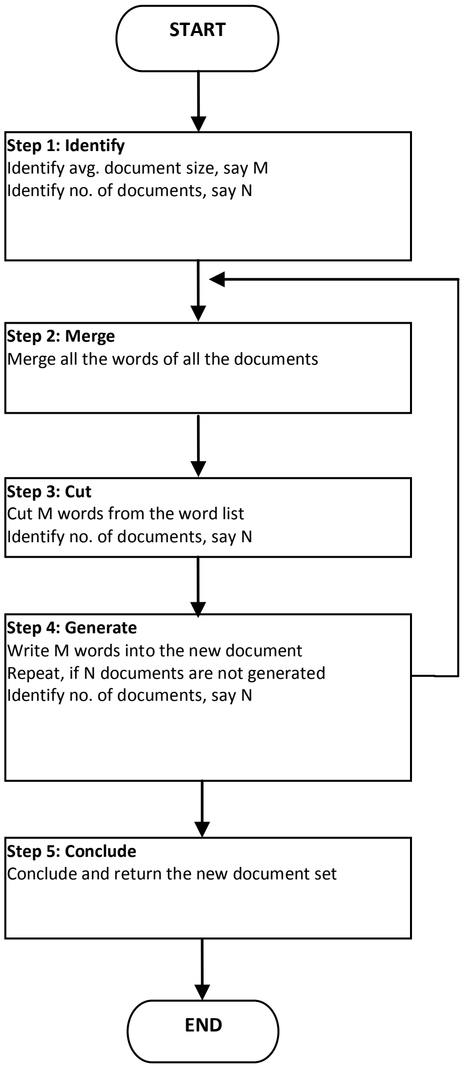

START
Step 1: Identify
Identify avg. document size, say M
Identify no. of documents, say N
Step 2: Merge
Merge all the words of all the documents
Step 3: Cut
Cut M words from the word list
Identify no. of documents, say N
Step 4: Generate
Write M words into the new document
Repeat, if N documents are not generated
Identify no. of documents, say N
Step 5: Conclude
Conclude and return the new document set
END

Fig. 2. Document Balancing

3. **Keyword Extraction Module:** To start with keyword extraction module, initially we add the context keywords along with a context factor i.e. $\mathbf{C_f}$ to the list of keywords. To create a dictionary of meaningful words from the set of documents, we present algorithm involving a factor which controls the number of false alarm i.e. NFA. If we observe that the word w appears m times in the same document, then we define this word as a meaningful word if and only if its NFA is smaller than some meaningfulness factor $\boldsymbol{\varepsilon}$. In other words, if the event of appearing m times has already happened, but the expected number is less than ε, we have a meaningful event. We also define a measure for the weight of meaningfulness to determine if the word w is indeed a meaningful word. The set of all meaningful words in a corpus of documents D_1,…., D_N, is defined as a set of meaningful keywords.

Let us also summarize how to generate the set of keywords K_W from a corpus of documents represented by set D. Let's assume that the document set D is pre-processed into parts represented by set P**.** For each word w taken from the document set P (P_1, …. P_N), perform the following steps:

1 Count the number of times K the word w appears in document set P (P_1,….,P_N)

2 For i from 1 to N

   i. Count the number of times m the word w appears in the document P_i

   ii. if m < 1, then skip to the next document in set P

   iii. Calculate NFA(w, P_i)

$NFA\ (\mathrm{w},\ \mathrm{P_i}) = \binom{K}{m}\frac{1}{N^{m-1}}$

iv. Calculate meaningfulness from the NFA, for some input context factor $c_f >= 0$

meaning (w, P_i) = $-\frac{1}{m}$log (NFA (w, P_i))+ $\mathbf{c_f} * \boldsymbol{\varepsilon}$

v. If meaning > $\boldsymbol{\varepsilon}$, then add word w to the set K_W and mark w as a meaningful word for P_i

We define a set of $\boldsymbol{\varepsilon}$ keywords as a set of all words with NFA< $\boldsymbol{\varepsilon}$, $\boldsymbol{\varepsilon}$ < 1. Smaller ε corresponds to more important words. It is easy to see that meaning(w, P)> $\boldsymbol{\varepsilon}$ is equivalent to NFA(w, P) < $\boldsymbol{\varepsilon}$. The $\boldsymbol{\varepsilon}$ is a parameter that is used to vary the size of the set typically chosen strictly positive as we are only interested in meaningful words.

4. **Keyword Index:** The meaningful keywords are identified by keyword extraction module. These keywords need to be stored for further reference and use by other modules. The index maintains calculated meaningfulness for these keywords to allow fast searches for comparisons during the post processing phase. It also maintains information about source of keywords such as originating document or user input.

5. **Small World Topology:** We are interested in graphs with a small world topology; the main reason is if a graph becomes a small world then we can reliably define the most important nodes and edges of such a graph by measuring their contributions to the graph being a small world. This gives us a mechanism to define the most important sentences and paragraphs. To define one parameter family of weighted graph Gr (D, $\boldsymbol{\varepsilon}$) for document set D, following steps are used.

1. Pre-process the document set D using the steps:

   i. Split the words by non-alphabetic characters.

   ii. Down-case all words

   iii. Apply stemming (lemmatization)

2. Let us denote S_1, S_2… S_n the sequence of consecutive sentences as vertexes for graph(D, $\boldsymbol{\varepsilon}$). Add an edge for every pair of consecutive sentences (S_i, S_{i+1}) with weight defined as ε .

3. Finally, if two sentences share at least one word from the set MeaningfulSet($\boldsymbol{\varepsilon}$) we connect them by the edge. The weight of the edge is determined from the value of meaningfulness. This defines the family of weighted graph Gr(D, $\boldsymbol{\varepsilon}$).

6. **Automatic Summarization:** In automatic summarization, we can think about the extracting of a small number of important sentences and paragraphs as removing a large number of unimportant sentences, whilst preserving the structure of a document. If we want to rank nodes according to some ranking function, it is important that this function gives us a wide range of values. We can define an extractive summary as follows**:**

1. Select a measure of centrality for small world networks. For example degree centrality, betweenness centrality, eigenvector centrality, closeness centrality etc.

2. Check that for the corresponding range of the parameter ε , this measure of centrality has a wide range of values and the heavy-tail distribution.

3. Select sentences and paragraphs with the highest ranking as a summary. We can also select the highest ranking paths in the graph if some coherence in the summary is desired.

7. **Post Processor:** Post processing phase mostly deals with presentation of the output set i.e. identified and ranked sentences. The output set is defined as the subset of sentences identified from input document set D.

$S_O \rightarrow \{S_{m1}, S_{m2} \ldots Smk\} \quad (S_O € \{S_1, S_2 \ldots S_n\})$

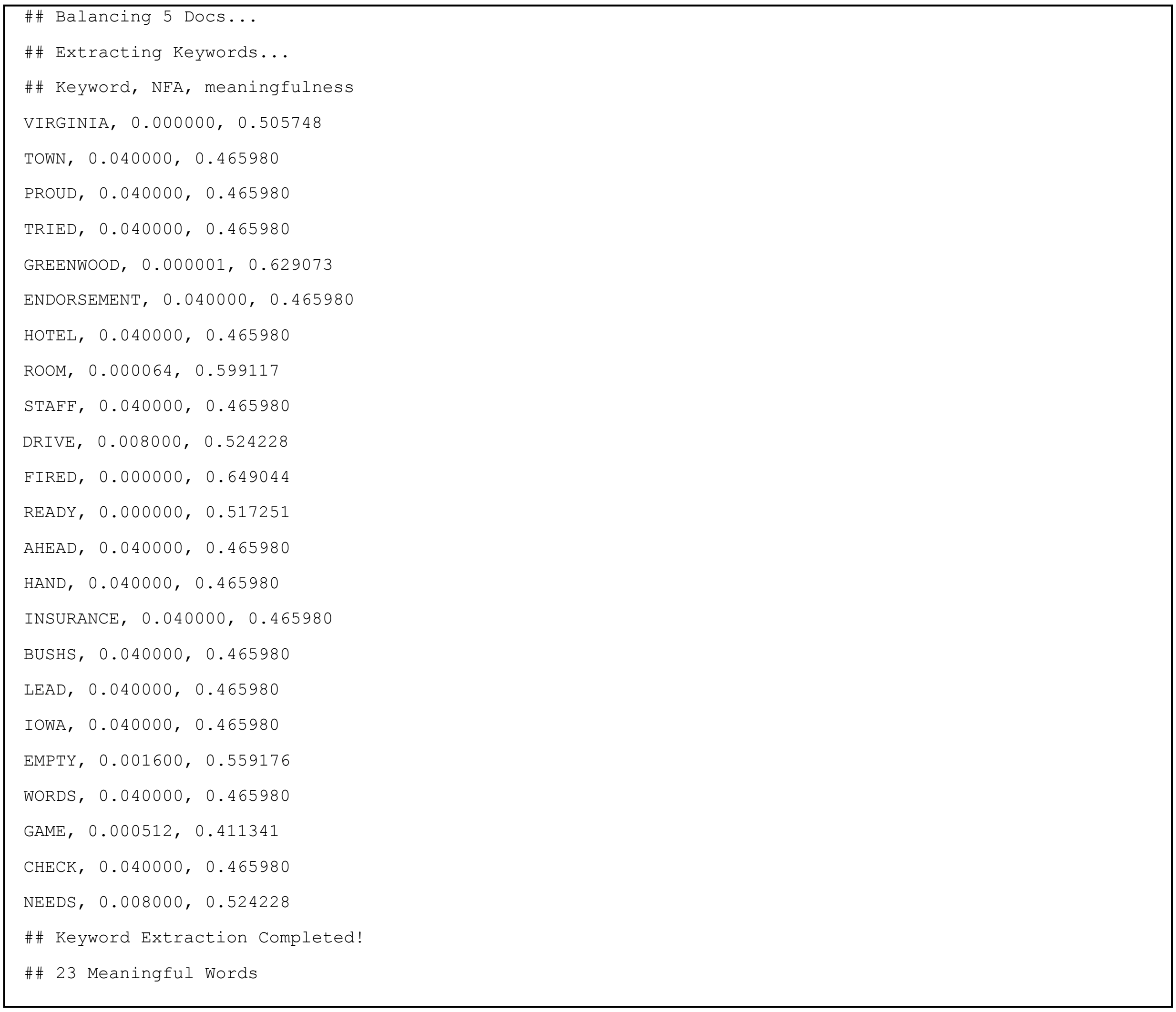
IV. Results

A. Data Set

A sample input was considered in the form of a set of 5 speeches by President Obama. We try to identify the meaningful words with N=5 and varying ε. The value of ε may be determined based on iterative selection and observation of the resulting performance, with reference to a specific context as input such as “economy”, “employment”, etc.]

B. Result Set

The output from a sample run of keyword extraction module on the input data set containing 5 documents with total 7473 words, which yields 23 meaningful keywords is shown below:



```
## Balancing 5 Docs...
## Extracting Keywords...
## Keyword, NFA, meaningfulness
VIRGINIA, 0.000000, 0.505748
TOWN, 0.040000, 0.465980
PROUD, 0.040000, 0.465980
TRIED, 0.040000, 0.465980
GREENWOOD, 0.000001, 0.629073
ENDORSEMENT, 0.040000, 0.465980
HOTEL, 0.040000, 0.465980
ROOM, 0.000064, 0.599117
STAFF, 0.040000, 0.465980
DRIVE, 0.008000, 0.524228
FIRED, 0.000000, 0.649044
READY, 0.000000, 0.517251
AHEAD, 0.040000, 0.465980
HAND, 0.040000, 0.465980
INSURANCE, 0.040000, 0.465980
BUSHS, 0.040000, 0.465980
LEAD, 0.040000, 0.465980
IOWA, 0.040000, 0.465980
EMPTY, 0.001600, 0.559176
WORDS, 0.040000, 0.465980
GAME, 0.000512, 0.411341
CHECK, 0.040000, 0.465980
NEEDS, 0.008000, 0.524228
## Keyword Extraction Completed!
## 23 Meaningful Words
```

Fig. 3. Output from sample run of Keyword Extraction algorithm

The experiments show that result set varies with variation in ε for specified input set. The selection of ε is based on the number of keywords that need to be selected. It is evident that smaller value of ε results in fewer keywords identified. However we observe that the size of meaningful set (ε) has a sharp drop in the number of meaningful words to 0, around some critical value as shown in the figure below:

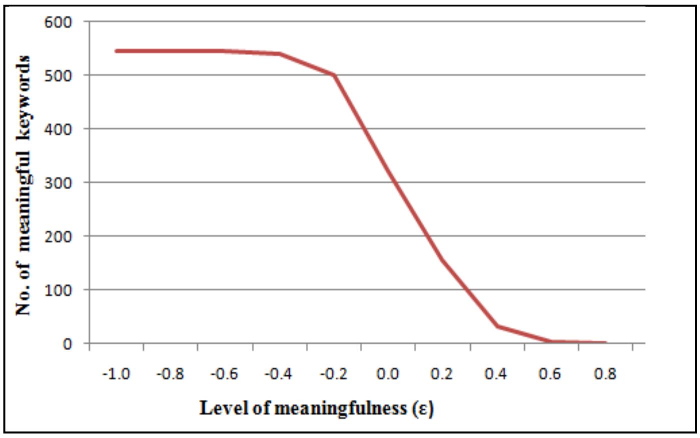


Fig. 4. Example of an image with acceptable resolution

V. Conclusion and Future Scope

The automatic keyword extraction module extracts meaningful keywords from the documents; the context supplied as input, influences the number of meaningful words. The documents then transform into a small world topology, with weighted graphs based on the value of meaningfulness. This helps in finding the important and relevant sentences in the documents. We can then define an extractive summary as a measure of centrality for small world networks. Select sentences and paragraphs with the highest ranking as a summary. We can also select the highest ranking paths in the graph if some coherence in the summary is desired. Thus we improve the results of automatic summary of documents which is relevant to the given context.

The Helmholtz technique allows fast processing of large amounts of data, and thus can be easily adapted to any type of data such as speeches, emails or messages. The experimental results of Helmholtz technique are lower than the ones of state-of-the-art summarizers for short documents, this method is both computationally very cheap and language independent. Weighted graphs help in improving the quality of summary. We would like to evaluate the performance and accuracy of such weighted graphs for a variety of data sets. Finally, increasing the target number of edges in the graph is also important to improve the results and can be dealt with in the future.

Acknowledgement

I express my deep sense of gratitude to Dr. D. D. Shah – Principle GHRCEM, Prof. Poonam Gupta – HOD of Computer Engineering, Prof. Vidya Dhamdhere M.E. coordinator for their timely guidance and cooperation. I am very much thankful to my guide Prof. B. Padmavathi for encouraging and motivating me in the right direction.

Causal Productions wishes to acknowledge Michael Shell and other contributors for developing and maintaining the IEEE LaTeX style files which have been used in the preparation of this template. To see the list of contributors, please refer to the top of file IEEETran.cls in the IEEE LaTeX distribution.

References

1. B Dadachev, A Balinsky, H Balinsky 2012 “On the Helmholtz principle for Data Mining”, Third International Conference on Emerging Security Technologies - IEEE.
2. Balinsky, H. Balinsky, and S. Simske, April 2011, “On the Helmholtz principle for data mining,” Proc. of 2011 Conf. on Knowledge Discovery, Chengdu, China.
3. G. Erkan, DR Radev 2004 –“LexRank: Graph-based Lexical Centrality as SalienceinText Summarization” J. Artificial Intelligence Research (JAIR).
4. H. Balinsky, A. Balinsky, and S. Simske, 2011 “Document sentences as a small world” Proc. of IEEE SMC.
5. Helen Balinsky, Alexander Balinsky and Steven J. Simske, 2011 "Automatic Text Summarization and Small-World Networks", ACM
6. H. Zha,“Generic summarization and key phrase extraction using mutual reinforcement principle and sentence clustering”, SIGIR2002.
7. J.M. Kleinberg. 1999. "Authoritative sources in a hyperlinked environment". Journal of the ACM, 46(5):604–632.
8. M. A. Hearst and D. Rosner. 2008, “Tag Clouds: Data Analysis Tool or Social Signaller? “,In Proc. of the 41st Hawaii Int. Conf. on System Sciences.
9. R. Mihalcea and P. Tarau, “TextRank: Bringing order into texts”, Proc. of the Conf. on Empirical Methods in NLP, pp. 404–411, 2004.

10. RM Alguliev, RM Aliguliyev 2009–"Evolutionary Algorithm for Extractive Text Summarization" Intelligent Information Management.
11. S. Brin and L. Page, "The anatomy of a large-scale hyper textual Web search engine", Computer Networks and ISDN Systems, 30(1–7), 1998.
12. Samer Hassan and Rada Mihalcea and Carmen Banea, 2007 "Random-Walk Term Weighting for Improved Text Classification" IEEE I International Journal of Semantic Computing.
13. Thomas Bohne Sebastian Rönnau Uwe M. Borghoff "Efficient Keyword Extraction for Meaningful Document Perception", 2011.
14. Xun Wang, Lei Wang, Jiwei Li, Sujian Li "Exploring simultaneous keyword and key sentence extraction: improve graph-based ranking using Wikipedia", ACM, 2012.
15. X. Wang, J. Yang, J. Xiao, "Towards an Iterative Reinforcement Approach for Simultaneous Document Summarization and Keyword Extraction", ACL2007.

AUTHOR(S) PROFILE

Mrs. Bhavana Lanjewar, is working as Assistant Professor in Computer Engineering Department, G.H. Raisoni Institute of Engineering and Technology. She completed Masters in Computer Engineering from Savitribai Phule Pune University. She has been teaching Computer Science for over 7 years Computer Science and her current research work includes data mining and information retrieval. She also has industry experience of over 3 years at various software organizations.