Extraction of Important Texts for Effective Multi Document Summarization Using Improved Fuzzy Logic

Patel Birva[1], Aakash Shah [2]
[1, 2] Dept of CE
[1, 2] Silver Oak College of Engineering & Technology, Gota, Ahmedabad, Gujarat, India

Abstract- *Text mining has different aspects to process sentences, words and texts. Many time it has become a lengthy and cumbersome process to understand and co-relate words and texts as part of sentences to generate a meaning which is in combination of these selected words.*

In this paper I have focused on advanced level text summarization where in extractive text summarization, important words and texts are selected based on certain important features which in turn extracting sentences containing it[1]. The importance of some extractive features is more than the some other features, so they should have the balance weight in computations.Based on which a graph is generated which has same weight balanced node at each level.The purpose of this paper is to use novel method and WorldNet dictionary features such as relative words and synonyms to handle the issue of ambiguity and imprecise values with the traditional two value or multi-value logic.
Methods like pre processing approach, match semantics, generate graph based on WorldNet synonyms will generate summary which will combine all the pages of one document[1]. Final summary is generated by selecting common sentences from all the different summaries generated each for one document set.

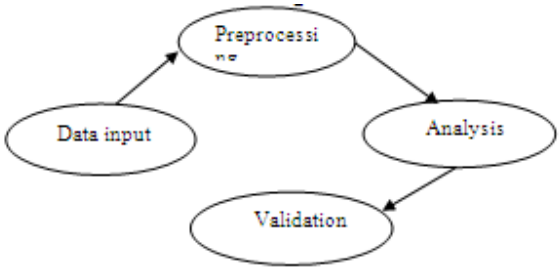
Keywords- Text summarization, WordNet, Text mining, Data mining

I. INTRODUCTION

Data mining is the computing process of discovering patterns in large involving methods at the intersection of machine learning, statistics, and database systems.

The **knowledge discovery in databases (KDD) process** is commonly defined with the[19]
stages:

(1) Selection
(2) Pre-processing
(3) Transformation
(4) Data mining
(5) Interpretation/evaluation[19].



TEXT MINING

Text mining, also referred to as ***text data mining***, roughly equivalent to **text analytics**, is the process of deriving high-quality information from text and sub type of data mining[20].

Typical text mining tasks include text categorization , text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling.

In today's world, the daily hustle-bustle does not permit a human being to devote time for manually summarizing various lengthy documents.[1]

Text Summarization produces a shorter version of large text documents by selecting most relevant information. Text summarization systems are of two types: extractive and abstractive[20].

Text Summarization plays an important role in the area of text mining and natural language processing. As the information resources are increasing tremendously, readers are overloaded with loads of information[3]. Finding out the relevant data and manually summarizing it in short time is much more difficult, challenging and tedious task for a human being[4].

Text summarization can be done by three different methods: fuzzy logic based method,bushy path method, and wordnet synonyms method are used to generate summaries[3]. Wordnet ontology is also used to generate abstractive summary from extractive summary.

Multi-document summarization aims to produce a compressed version of numerous online text documents and preserves the salient information[1].

A particular challenge for multi-document summarization is that there is an inevitable overlap in the information stored in different documents[20].

Multi-document summarization is useful when a user deals with a group of heterogeneous documents and wants to compile the important information present in the collection, or there is a group of homogeneous documents, taken out from a large corpus as a result of a query[2].

This paper introduces a clustered genetic semantic graph approach for multi-document abstractive summarization. The semantic graph from the document set is constructed in such a way that the graph vertices represent the predicate argument structures The clustering algorithm is performed to eliminate redundancy in such a way that representative PAS with the highest salience score from each cluster is chosen, and fed to language generation to generate summary sentences[20].

This paper proposes an innovative graph-based text summarization model for generic single and multi-document summarization. The approach involves four unique processing stages: parsing sentences semantically using Semantic Role Labeling (SRL), grouping semantic arguments while matching semantic roles to Wikipedia concepts, constructing a weighted semantic graph for each document and linking its sentences (nodes) through the semantic relatedness of the Wikipedia concepts[6].

The contributions of this paper are as follows.

1)we avail SRL-based semantic representation of sentences to group similar arguments from each role-set and project them onto corresponding Wikipedia concepts[4].
2) we propose a weighted semantic document graph where each sentence is represented by the sub-nodes containing the concepts of its semantic arguments[4]. The semantic relatedness between the Wikipedia concepts of the semantic arguments forms the edge-weights[4].
3) The performance of our summarizer is
empirically validated using the standard DUC2002 dataset[4].

In each cluster, each sentence is assigned five different weights

1. Chronological weight of sentence (Document level)
2. Position weight of sentence (position of sentence in the document)
3. Sentence weight (based on term weight)
4. Aspect based weight (sentence containing aspect words)
5. Synonymy and Hyponym Weight.

Then top ranked sentences having highest weight are extracted from each cluster and presented to user.
APPLICATIONS

- Tracking and Summarizing news articles on a daily-basis.
- Summarizing Chapters from various reference books.
- Speech Summarization

II. RELAED BACKGROUND

A. Text Mining Process

Following step for Text mining process

Step 1: Text Preprocessing

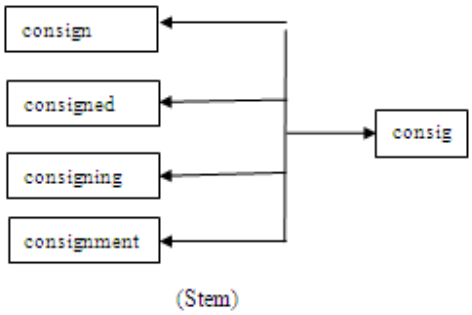
Text preprocessing is the initial step of text mining which reads one text document at time and processes it. This step divides into following main three subtasks [4]

1.1 Tokenization

Tokenization is the process of breaking a stream of textual content up into words, terms, symbols, or some other meaningful elements called tokens[21].The list of tokens turns into input for in additional processing including parsing or Text Mining[21]**.**

1.2 Stemming

In linguistic morphology and information retrieval, **stemming** is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form—generally a written word form[22]. The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem[22].



1.3 Stop Word Elimination

In computing, stop words are words which are filtered out before or after processing of natural language data (text). Though "stop words" usually refers to the most common words in a language, there i no single universal list of stop words used by all natural language processing tools, and indeed not all tools even use such a list[23]. Some tools specifically avoid removing these stop words to support phrase search.

1.4 Graph Selection Method

The semantic graph from the document set is constructed in such a way that the graph vertices represent the predicate argument structures The clustering algorithm is performed to eliminate redundancy in such a way that representative PAS with the highest salience score from each cluster is chosen, and fed to language generation to generate summary sentences[3].

Word net is used to find the semantic roles in a sentence and using these roles conceptual Graph is constructed Raw text are pre-processed and disambiguated nouns are mapped to Wordnet concepts[24]. Concept rather than words are very efficient, concise representation of document content. It can easily and clearly interpretable. Co-occurrence of concepts rather than words together is calculated on the basis if hypernym and holonym occurrence together. Page rank is used to infer the correct sense of concept in the document[3].

1.5 Keyword Extraction

Keyword extraction is tasked with the automatic identification of terms that best describe the subject of a document[25].

Key phrases, *key terms*, *key segments* or just *keywords* are the terminology which is used for defining the terms that represent the most relevant information contained in the document[25].

B. Tool

2.1 RapidMiner

RapidMiner is a data science software platform developed by the company of the same name that provides an integrated environment for machine learning, deep learning, text mining, and predictive analytics[26]. It is used for business and commercial applications as well as for research, education, training, rapid prototyping, and application development and supports all steps of the machine learning process including data preparation, results visualization, validation and optimization[26].

RapidMiner is developed on an open core model. The RapidMiner (free) Basic Edition, which is limited to 1 logical processor and 10,000 data rows, is available under the AGPL license. [26]

C. Dictionary

2.2 WordNet

WordNet is a lexical database for the English language. It groups English words into sets of synonyms called *synsets*, provides short definitions and usage examples, and records a number of relations among these synonym sets or their members[18].

WordNet can thus be seen as a combination of dictionary and thesaurus. While it is accessible to human users via a web browser, its primary use is in automatic text analysis and artificial intelligence applications.

The database and software tools have been released under a BSD style license and are freely available for download from the WordNet website. Both the lexicographic data (*lexicographer files*) and the compiler (*called grind*) for producing the distributed database are available[18].

WordNet includes the lexical categories nouns, verbs, adjectives and adverbs but ignores prepositions, determiners and other function words[18].

III. PROPOSED SYSTEM

3.1 Proposed Flow

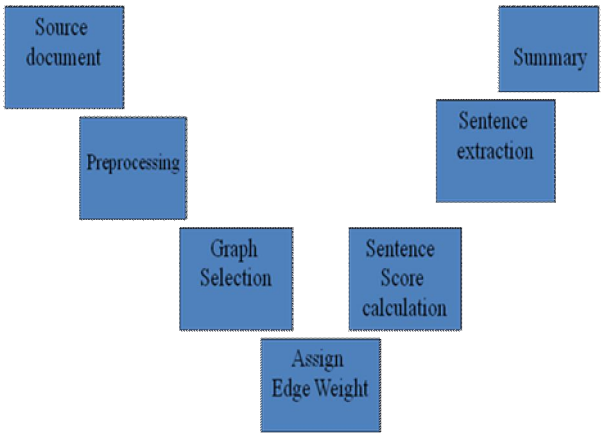


Fig 1 Proposed flow

3.2 Algorithm

Step 1.For each document

Step 2.With Term T and Sentence S.
[Compute word frequency in Document by traversing Sentences]

Step 3.Extraction of word in to token for each doc
[In this step tokenization will be done..]

Step 4.Stop words elimination process of extracted words (using classic method) In this step elimination of unwanted words and redundant words would be eliminated..

Step 5.Traverse each words and calculate its frequency & store them in temp_doc

TF = (N * (No. of Document/No. of Document where keyword occurred)/IDF

N = Number of time word occurs in current Document

IDF = log_e (No. of Document/No. of Document where keyword occurred)

[Frequency would be calculated of words which are used many times using Term Frequency and inverse document Frequency methods.]

Step 6.End For

Step 7.For each temp_doc
[For all documents all steps are performed as above.]

Step 8.Perform Graph selection algorithm using [WordNet Using wordnet graph selection algorithm is performedand weight to the wordis assigned.]

Step 9.Store the Term, Frequency, Position in document of sentence in a Doc in Vector.
[Keywords are stored in a vector and it is stored in temporary file.]

Step 10.Stemming on file to removing similar meaning words.
[Removal of words with similar meanings is done in this steps.And redundant words would be deleted.]

Step 11. End for

Step 12.Select top N Keywords from vector
[Select n keywords to create summary from a vector.]

Step 13.Extract Noticeable Keywords.
[Extract noticeable keywords from the keywords.]

Step 14.Form Summary
[Final summary is generated in this step.]

IV. EXPERIMENT & RESULT

Document can be of any length in size and can have any type of data(numeric or textual but not images) ,our approach can summarize multiple documents .
Based on NLP we can extract most important words first and than only similar sentences and similar words.

The target is to find final of words & text, which are individual and unique.

Target is to get better recall ,precision & f-measure results with higher accuracy.

Auto text summarization

-Multi document
-Fuzzy information
-Automatic process (without user intervention)

Table 1 proposed flow result

Accuracy	Precision	Recall
85.02%	86.92%	84.00%
86.94%	87.33%	85.17%

Now, we see to obtain the result form proposed flow that yields the best results. In Table 1 show the result of precision and recall for proposed flow output. In training dataset, Proposed Flow achieves accuracy 85.02% when the precision is 85.92% and recall 84%. In testing dataset, Proposed Flow achieves accuracy 86.94% when the precision is 87.33% and recall 85.17%.

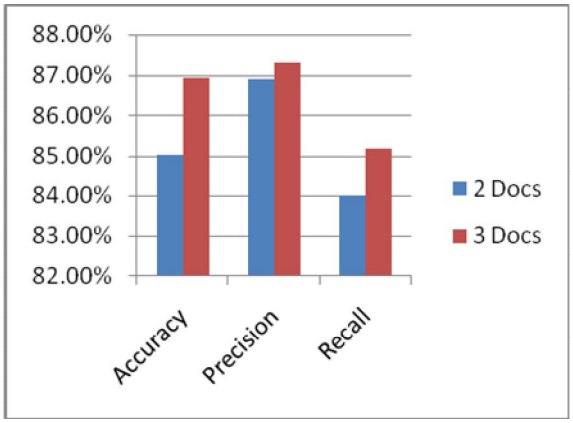


Fig 2 proposed flow result

Furthermore, another experiment is carried out to obtain the number of fold with the best result. The result is shown as Fig.2. We can see from the graph that, when the result reaches the highest point.

Table 2 Documents with different word limits

Word limit	Summary Generated	Percentage
20,000	2167	10.83%
40,000	9243	23.10%
60,000	12638	21.06%

Here the table shown below is consisting of information about the summary generated using different different documents of having word limits respectively 20000,40000 and 60000.generated summary is consist of words respectively 2167,9243 and 12638 words and the percentage of summary.this results are calculated using precision,recall and accuracy equations and term frequency and inverse document frequency formulas.

V. CONCLUSION

Our main goal in this paper is to find the best featured sentence that could impact to form the summary. For these, we have created cluster for each different weigh word from all documents. Apart form text pre processing, we have introduced feature based selection that help in ranking the sentences and to get the best sentences from each vector. The summary generated by the proposed algorithm follows the extraction method, when it finds the most unique sentences based on each word. it is containing that one sentence contains a formal person, place or thing.

We are dealing with conceptual synopsis in which we are thinking to combine sentences to make another single sentence.

REFERENCES

[1] Jyoti Yadav & Dr. Yogesh Kumar Meena, "**Use of fuzzy logic and wordnet for improving performance of extractive automatic text summarization**" 2016,IEEE, 978-1-5090-2029-4

[2] Harsha Dave & Shree jaswal "**Multiple text document summarization system using hybrid summarization technique"** 2015,IEEE, 978-1-4673-6809-4

[3] Atif Khan, Naomie Salim & Haleem Farman "**Clustered genetic semantic graph approach for multi-document abstractive summarization"** 2016, IEEE(copy right), 978-1-4673-8753-8

[4] Muhidin Mohamed & Mourad Oussalah, " **An iterative graph-based generic single and multi document summarization approach using semantic role labeling and wikipedia concepts"** 2016, IEEE, 978-1-5090-2251-9

[5] Deepak Sahoo , Rakesh Balabantaray , Mridumoni Phukon & Saibali Saikia , "***Aspect Based Multi-Document Summarization"*** 2016, IEEE (copyright), 978-1-5090-166

[6] Amol Tandel, Brijesh Modi, Priyasha Gupta, Shreya Wagle & Mrs. Sujata Khedkar, **"Multi-document text summarization - A survey"** 2016,IEEE

[7] A. Khan, N. Salim, and Y. J. Kumar, "A framework for multi-document abstractive summarization based on semantic role labelling," *Applied Soft Computing*, vol. 30, pp. 737-747, 2015.

[8] Y. J. Kumar, N. Salim, A. Abuobieda, and A. Tawfik, "Multi document summarization based on cross-document relation using voting technique," in *Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on*, 2013, pp. 609-614.

[9] A. Khan, N. Salim, and Y. J. Kumar, "A framework for multi-document abstractive summarization based on semantic role labelling," *Applied Soft Computing*, vol. 30, pp. 737-747, 2015.

[10] Y. J. Kumar, N. Salim, A. Abuobieda, and A. Tawfik, "Multi document summarization based on cross-document relation using voting technique," in *Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on*, 2013, pp. 609-614.

[11] D. Das and A. F. Martins, "A survey on automatic text summarization," *Literature Survey for the Language and Statistics II course at CMU*, vol. 4, pp. 192-195, 2007.

[12] H. P. Luhn, *The Automatic Creation of Literature Abstracts*, IBM Journal of Research and Development, 2(2), 159-165, 1958.

[13] P. B. Baxendale, *Machine-made Index for Technical Literature: An Experiment*, IBM J. Res. Dev., 2(4), 354-361, 1958.

[14] H. P. Edmundson, *New Methods in Automatic Extracting*, J. ACM, 16(2), 264-285, 1969.
[15] Pragnya Addala, Text Summarization A Literature Survey , https://www.scribd.com/doc/235008952/Text-Summarization-Literature-Surveyscribd,on Jul 24, 2014
[16] Amit.S.Zore1, Aarati Deshpande, Extractive Multi Document Summarizer Alogorithim, In (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5245-5248 ISSN:0975-9646
[17] Nenkova, Ani, Sameer Maskey, and Yang Liu. "Automatic summarization."Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011. Association for Computational Linguistics, 2011.
[18] http://wordnet.princeton.edu
[19] http://www.techopedia.com
[20] http://en.m.wikipedia.org
[21] https://searchsecurity.techtarget.com
[22] https://nlp.stanford.edu
[23] https://www.packtpub.com
[24] https://link.springer.com
[25] https://www.airpair.com
[26] https://en.wikipedia.org/wiki/RapidMiner