IMPROVE THE QUALITY OF IMPORTANT SENTENCES FOR AUTOMATIC TEXT SUMMARIZATION

Michael George

Department of Information Technology, Dubai Municipality, Dubai City, UAE

ABSTRACT

There are sixteen known methods for automatic text summarization. In our study we will use Natural language processing NLP within hybrid approach that will improve the quality of important sentences selection by thickening sentence score along with reducing the number of long sentences that would be included in the final summarization. The based approach which is used in the algorithm is Term Occurrences.

KEYWORDS

Text summarization, Data mining, Natural language processing, Sentence scoring, Term Occurrences.

1. AIM

This research will provide an algorithm to improve important sentences quality for automatic text summarization. This method suitable for search engines, business intelligence mining tools, single document summarization and filtered summarization that rely on the top short list of important sentences.

2. INTRODUCTION

Automatic Text summarization [1] is a mechanism of generating a short meaningful text that can summarize a textual content by using computer algorithm. The quality of the summarization depends on the quality of the selection ability of the important sentences, paragraphs out of the main document that was given as input. That list of the sentences will be used in the formation of the final summarization with different custom ways which will represent the original content.

Automatic Text summarization is sub method in Data mining. And it is necessary in many sectors such as Search engines, Education, Business intelligence, Social media, and e-commerce.

As one of artificial intelligence functions, automatic text summarization have sensitive operations that require accuracy for meanings capturing, since there is no awareness to understand the

Natarajan Meghanathan et al. (Eds) : CSITA, DMAP, ISPR, NMCO - 2017
pp. 09– 15, 2017. © CS & IT-CSCP 2017 DOI : 10.5121/csit.2017.71402

content. Recently as data became large enough that makes easy classification is big challenge, while natural language processing NLP [2] tools advancing and as it's the backbone for text summarization, approaches and algorithms been developed to reform and improve the quality of the output. It become necessary to shorthand the data with the most important content text summarization is an efficient method for knowledge mining and extraction for many different sector.

Text Summarization Methods and approaches which currently in Development such as Neural networks [5], Graph theoretic [6], Term Frequency-Inverse Document Frequency (TF IDF) [7][8], Cluster based [8], Machine Learning [9], Concept Oriented [10], fuzzy logic [11][12][13], Multi document Summarization[14][15], Multilingual Extractive [16][17].

We are addressing the techniques that improves term occurrence processing that gives better score for the sentences selected to be included in the summarization.

3. RELATED WORK

3.1. Term Frequency-Inverse Document Frequency (TF IDF) approach

TF IDF stands for term frequency-inverse document frequency, and the tf-idf [7][8][3] weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the tf-idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query.

TF: Term Frequency, which measures how frequently a term occurs in a document. Since every document is different in length, it is possible that a term would appear much more times in long documents than shorter ones. Thus, the term frequency is often divided by the document length, the normalization equation:

$$\mathrm{tf}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Where ni,j is the number of occurrences of the considered term (ti) in document dj, divided by ∑k nk, j which is the total number of words in document dj.

IDF: Inverse Document Frequency is a measure of how much information the word provides, that is, whether the term is common or rare across all documents. It is the logarithmically scaled inverse fraction of the documents that contain the word, obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient as the following equation:

$$\mathrm{idf}_i = log \frac{|D|}{1 + |\{j : t_i \in d_j\}|}$$

Where |D| : total number of documents in the corpus divided by {j: ti ∈ dj}: number of documents where the term t appears, if the term is not in the corpus, this will lead to a division by zero. It is therefore common to adjust the denominator to 1 + {j: ti ∈dj} and in that cases which the inputs is a single document only, then IDF value will = 1.

Then we have the value of TF-IDF as TF * IDF for each term.

By sorting the results descending, we will get the highest terms in the given inputs.

3.2. Important sentences using Term Frequency

Usually the evaluation of sentence score, equal the sum of total terms value in the sentence, first split the document into sentences and counting the score for each sentence then sort them down descending to get the top sentences which would include in the summarization. Terms denseness gives the sentence its score.

4. THE METHODOLOGY AND ALGORITHM

We developed an approach we called it thick sentences that thickening sentence score which is Re-filtering important sentences, to fetch sentences with higher value and density and lower length. In short it's summarizing the summarization.

Our target is increasing sentence value along with reducing unnecessary words and long sentences, which make the top sentences list has more value. So that sentence score equal total sentence terms occurrence divided by sentence words, as the following equation:

$$SS = \frac{\sum TOC}{SW_n}$$

Where SS is the total sentence score and ∈ TOC (term occurrence count) is sum of sentence terms occurrence (number of appearance in the whole document\s), SWn is the sentence words count.

4.1. Data Extraction and procedure steps

- **Sentences splitting**: split the inputs (document/s) into array of sentences to improve terms fetching.

- **Natural Language Processing NLP**: in this part we used NLP Tools [4] to extract the keywords from the each sentence which have meaningful value such as nouns and adjectives, excluding stop words which can give false result. In term frequency approach, terms extraction happens once and globally for the entire inputs and getting high terms by filtering the top terms based on their frequency. In our method terms extraction happens for each sentence separately to reduce the time for terms comparison and avoiding missing terms.

- **Loop and indexing**: each sentence had index with the array of terms related, indexing the sentence to define its score based on evaluation result.

- **Score Evaluation**: applying our method on each sentence loop to get each score from related terms, as per the formula "SS = ∈TOC / SWn".

- **Promote important sentences**: sort the results based on score descending and get top sentences.

4.2. Algorithm

```
_____________________________________________________
N = the number of all sentences in the entire input
FOR  i = 0 To N
      SS = 0 'sentence score'
      TOC = 0 'terms Occurrence'
      ST = List of terms for (N -i)

      FOR  j = 0 To ST
         TOC = TOC + (ST-j n) 'term Occurrence'
      END LOOP

      SS = TOC / (N -i)Wn

      IF SS > 0 THEN
         AddToImportantSentencesList(N-i 'Current Sentence', SS)
END LOOP
_____________________________________________________
```

Algorithm Output was the list of important sentences sorted descending by score. The method shows good results if the target was fetching a short number of top sentences.

5. RESULTS

We tested our method on Science-space articles from (20 newsgroups datasets) [18], in the following table the results of top ten sentences in comparison with the initial term frequency method, sorted by our score descending.

As the both approaches are dealing with terms occurrence to promote the important sentences, the advantage here that we reduced results length, without losing the value of meaning or terms.

As we can see in "Table 1" we are getting short sentence in the top along with high terms occurrence by sorting by our score.

Where the sentences are contain strong related terms in most of its words.

Table 1. Score comparison for 20 newsgroups dataset.

Sentence.	Words Count.	TF Score	Our Score.
1st	11	1.06	7.91
2nd	6	0.55	7.83
3rd	30	2.83	7.73
4th	9	0.82	7.67
5th	12	1.08	7.58
6th	21	1.90	7.52
7th	9	0.82	7.44
8th	38	2.91	5.95
9th	16	1.20	5.88
10th	25	1.80	5.88

6. CONCLUSIONS

Clean selection of important sentences is the first stage to have efficient summarization. Sentence score is in making by many approaches, term frequency is one of the best methods from important sentences detection. By using thick sentences we have better quality as we can see from the following points.

- Promote short sentences which has a high terms occurrence.
- Unload unnecessary words which can increase the final summarization.
- Fast implementation from logic side.
- Summarize the summarization.

In this study we reviewed the main methods and ways to summarize text, we improved the term occurrence method by thickening sentence score and our system used C# 4.0 framework and Apache OpenNLP [4].

7. FUTURE WORK

Our equation was developed for specific scope that will be used mostly for web content.

The challenge when we tested the same approach but with larger inputs which contains a huge number of documents such as large books, then the results had less quality as term occurrence will not give accurate values and therefore the equation need to be adapted to cover this gap.

8. SUPPLEMENTARY MATERIAL

As it is about text summarization, so we used our method to summarize our paper itself and include top 5 sentences here as secondary visual result in the following Table.

Table 2. top 5 important sentences in the current paper

Sentence	Score
automatic text summarization is sub method in data mining	6.33
terms denseness gives the sentence its score	5.71
promote important sentences: sort the results based on score descending and get top sentences	5.33
this research will provide an algorithm that improves important sentences quality for automatic text summarization	4.88
sentence score is in making by many approaches, term frequency is one of the best methods from important sentences detection	4.5
this method suitable for search engines, business intelligence mining tools, single document summarization and filtered summarization that rely on the top short list of important sentences	4.31

ACKNOWLEDGEMENTS

I would like to Thank all those who Supported and encourage me, and for whom helped on that research, also thanks to Dubai Municipality for the good environment, tools and support from great management.

REFERENCES

[1] Aarti Patil, Komal Pharande, Dipali Nale, Roshani Agrawal "Automatic Text Summarization" Volume 109 – No. 17, January 2015.

[2] Ronan Collobert, “Natural Language Processing (Almost) from Scratch” 2011.

[3] TF-IDF "term frequency-inverse document frequency" tfidf.com.

[4] NLP "Apache OpenNLP" opennlp.apache.org.

[5] Khosrow Kaikhah, "Automatic Text Summarization with Neural Networks", in Proceedings of second international Conference on intelligent systems, IEEE, 40-44, Texas, USA, June 2004.

[6] G Erkan and Dragomir R. Radev, “LexRank: Graph-based Centrality as Salience in Text Summarization”, Journal of Artificial Intelligence Research, Re-search, Vol. 22, pp. 457-479 2004.

[7] Joeran Beel "Research-paper recommender systems: a literature survey" November 2016, Volume 17, Issue 4, pp 305–338.

[8] KyoJoong Oh "Research Trend Analysis using Word Similarities and Clusters" Vol. 8, No. 1, January, 2013.

[9] Joel larocca Neto, Alex A. Freitas and Celso A.A.Kaestner, "Automatic Text Summarization using a Machine Learning Approach”, Book: Advances in Artificial Intelligence: Lecture Notes in computer science, Springer Berlin / Heidelberg, Vol 2507/2002, 205-215, 2002.

[10] Meng Wang, Xiaorong Wang and Chao Xu, "An Approach to Concept Oriented Text Summarization", in Proceedings of ISCIT’05, IEEE international conference, China,1290-1293, 2005.

[11] Farshad Kyoomarsi, Hamid Khosravi, Esfandiar Eslami and Pooya Khosravyan Dehkordy, "Optimizing Text Summarization Based on Fuzzy Logic", In proceedings of Seventh IEEE/ACIS International Conference on Computer and Information Science, IEEE, University of Shahid Bahonar Kerman, UK, 347-352, 2008.

[12] Ladda Suanmali, Mohammed Salem, Binwahlan and Naomie Salim, "Sentence Features Fusion for Text summarization using Fuzzy Logic, IEEE, 142-145, 2009.

[13] Ladda Suanmali, Naomie Salim and Mohammed Salem Binwahlan, "Fuzzy Logic Based Method for Improving Text Summarization", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 2, No. 1, 2009.

[14] Junlin Zhanq, Le Sun and Quan Zhou, "A Cue-based HubAuthority Approach for Multi-Document Text Summarization", in Proceeding of NLP-KE'05, IEEE,642- 645, 2005.

[15] Chin-Yew Lin and Eduard Hovy," From Single to Multidocument Summarization: A Prototype System and its Evaluation", Proceedings of the ACL conference, pp. 457–464. Philadelphia, PA. 2002.

[16] David B. Bracewell, Fuji REN and Shingo Kuriowa, "Multilingual Single Document Keyword Extraction for Information Retrieval", Proceedings of NLP-KE'05, IEEE, Tokushima, 2005.

[17] Dragomir Radev "MEAD - a platform for multi document multilingual text summarization", In Proceedings of LREC 2004, Lisbon, Portugal, May 2004.

[18] 20 newsgroups, "Naive Bayes algorithm for learning to classify text" cs.cmu.edu.

AUTHOR

Michael George Girgis, born in Cairo Egypt, 1987,
A Software engineer, specialist in Data mining and machine learning algorithms
Has a Bachelor degree in Management Information systems,
From Obour Academy Cairo, Egypt.
Interested in Addressing Association and text Analysis Algorithms.