Automatic Extractive Text Summarization Using TF-IDF

WONG Hok Fong, WU Hengyu
The Chinese University of Hong Kong
{silaswhf,henrywu21}@link.cuhk.edu.hk

Abstract

Automatic text summarization is currently one of the most widely accepted techniques to get a concise and brief outline of a piece of text. To extract sentences for a summary, the Term Frequency-Inverse Document Frequency (TF-IDF) Algorithm is applied to give a hierarchy for sentences. This report serves to introduce several key concepts of the TF-IDF algorithm through probabilistic method and its resemblance to information theory and evaluates its robustness in text summarization.

1 Introduction

While abstractive text summarization may require semantic understanding of the text, extractive text summarization works by extracting and isolating key information from a pre-existing text, compressing the text into a summarized version. The concepts of TF-IDF are held responsible for ranking sentences.

2 On the theoretical arguments of TF-IDF

2.1 Background

The Term Frequency-Inverse Document Frequency (TF-IDF) Algorithm is originally developed for information retrieval and text mining. Given a query string, the algorithm assign scores, ranks and then selects the most relevant document from a collection of texts.

2.2 Motivation

Assume that there are D documents in a database (called "corpus", as depicted in Figure 1).

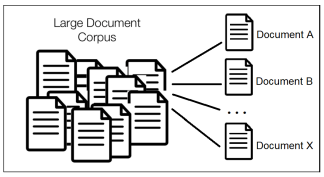


Figure 1: The concept of corpus and document.

For a query string, the document which contains more occurrence of a specific word should be more relevant than that contains less. This is the idea of term frequency (TF).

However, the term frequency is not sufficient to measure word importance. Consider the following case where in the Brown Corpus of American English text, the word "the" is the most frequently

occurring word, and by itself accounts for nearly 7% of all word occurrences ($69,971$ out of slightly over 1 million). However, "the" does not bear significance in the importance of documents.

In 1972, Karen Spärck Jones published a paper in the *Journal of Documentation* called "A statistical interpretation of term specificity and its application in retrieval", with the insight "the specificity of a term can be quantified as an inverse function of the number of documents in which it occurs" (Sparck Jones, 1972). This is the precursor of inverse document frequency (IDF).

2.3 Empirical approach

Assumption 1 (Statistical Distribution of a Word) We can consider the probability that a random document D_i within a D-document corpus would contain the term w with estimation of

$$\mathbb{P}(w) = \mathbb{P}(w \text{ occurs in } D_i) \approx \frac{d_w}{D},$$

where d_w depicts that term w occurs in d_w documents in the corpus.

Assumption 2 (Zipf's Law) Frequency decreases very rapidly with rank, to the extent that the log of frequency against the log of rank is a linear function (as illustrated in Figure 2).

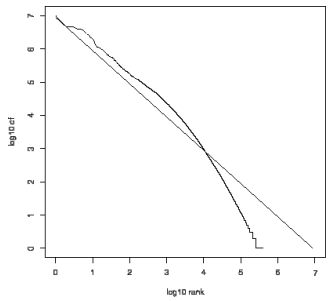


Figure 2: Zipf's law depicting log of frequency against the log of rank (Manning et al., 2008).

We can then reasonably define IDF in terms of probability, and

$$IDF(w) = -\log \mathbb{P}(w).$$

Combining the concept of TF, we have

$$\text{TF-IDF} = \frac{n_w}{N} \log \frac{D}{d_w}.$$

2.4 Information theory approach

Definition (Entropy in Information Theory) Entropy is defined as the average level of "information", "surprise", or "uncertainty" inherent to the variable's possible outcomes. Given a discrete random variable X, which takes values in the alphabet $\mathcal{X}$ and is distributed according to $p : \mathcal{X} \to [0, 1]$:

$$H(X) = \sum_{x \in \mathcal{X}} -p(x) \log p(x)$$

depicts the expected value of the self-information of a variable.

Assumption 1 (Document Size) Suppose there are D documents in the corpus, and the size of each document is the same, all with M words. In total, there would be $N = MD$ words in the corpus. Then,

$$M = \frac{N}{D} = \frac{\sum_w n_w}{D},$$

where n_w denotes the number of occurrence of the word w in the corpus.

Assumption 2 (Word Importance Contribution) Only the existence of a word w in a document will be considered, no matter how many times it occurs in a document (denoted as d_w). Then, either a word w constitutes a contribution of $c_w = \frac{n_w}{d_w}$, or it has no contribution at all.

Noting that $c_w < M$, we have

$$I(w) = n_w \log \frac{N}{n_w} = n_w \log \frac{MD}{c_w d_w} = n_w \log \frac{D}{d_w} \frac{M}{c_w}.$$

We can also normalize all words' $I(w)$ and therefore $I(w) = TF(w) \log \frac{D}{d_w} \frac{M}{c_w}$.

Comparing to the TF-IDF score: $TF(w) \cdot IDF(w) = \frac{n_w}{N} \log \frac{D}{d_w}$,

we have

$$\text{TF-IDF} = I(w) - TF(w) \log \frac{M}{c_w}.$$

If the word w carries more information with higher $I(w)$, it has the higher TF-IDF value; at the same time the higher frequency of w in documents where w has its existence would enforce the second term to be smaller. These facts are in reconciliation with information theory.

3 The application of TF-IDF on text summarization

3.1 Explanation

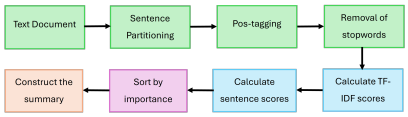
Since there is only one piece of text for extraction, it will be important to transcend the concepts. Taking each sentence in the text as "documents", and the text with set of sentences to be the "corpus", we assign the importance to a sentence by the sum of TF-IDF values of each word.

Formally, let sentence S be a tuple $(w_1, w_2, \cdots, w_M)$, the importance is then defined as

$$\sum_{k=1}^{M} \text{TF-IDF of } w_i = \sum_{k=1}^{M} TF(w_i) \cdot IDF(w_i).$$

Then, the sentence with larger importance should rank higher, and should be more likely to be retained.

3.2 Program workflow



3.3 Demonstration

Sample Text: Women education is a catch all term which refers to the state of primary, secondary, tertiary and health education in girls and women. There are 65 Million girls out of school across the globe; majority of them are in the developing and underdeveloped countries. All the countries of the world, especially the developing and underdeveloped countries must take necessary steps to improve their condition of female education; as women can play a vital role in the nation's development. If we consider society as tree, then men are like its strong main stem which supports the tree to face the elements and women are like its roots; most important of them all. The stronger the roots are the bigger and stronger the tree will be spreading its branches; sheltering and protecting the needy. Women are the soul of a society; a society can well be judged by the way its women are treated. An educated man goes out to make the society better, while an educated woman; whether she goes out or stays at home, makes the house and its occupants better. Women play many roles in a society- mother, wife, sister, care taker, nurse etc. They are more compassionate towards the needs of others and have a better understanding of social structure. An educated mother will make sure that her children are educated, and will weigh the education of a girl child, same as boys. History is replete with evidences, that the societies in which women were treated equally to men and were educated; prospered and grew socially as well as economically. It will be a mistake to leave women behind in our goal of sustainable development, and it could only be achieved if both the genders are allowed equal opportunities in education and other areas. Education makes women more confident and ambitious; they become more aware of their rights and can raise their voice against exploitation and violence. A society cannot at all progress if its women weep silently. They have to have the weapon of education to carve out a progressive path for their own as well as their families.

Summary (retaining 20% of the sentences): An educated man goes out to make the society better, while an educated woman; whether she goes out or stays at home, makes the house and its occupants better. Women play many roles in a society- mother, wife, sister, care taker, nurse etc. An educated mother will make sure that her children are educated, and will weigh the education of a girl child, same as boys.

Women play many roles in a society- mother, wife, sister, care taker, nurse etc.

Term	TF	IDF	Term	TF	IDF
woman	0.071428	0.176091	sister	0.071428	1.176091
play	0.071428	0.875061	care	0.071428	1.176091
role	0.071428	0.875061	taker	0.071428	1.176091
society	0.071428	0.397940	nurse	0.071428	1.176091
mother	0.071428	0.875061	etc	0.071428	1.176091
wife	0.071428	1.176091			

Figure 3: The most important sentence and its TF-IDF values.

Although the term “woman” has high frequency in the corpus, it does not appear to have high TF-IDF value (due to its low specificity). Instead, the term “child” only appears twice in the corpus and has rather high mean TF-IDF value, thus is shown in the summarization. (See **Appendix** for the graph.)

4 Improvements

The project considers several possible improvements.

1. Words having the same root are not identified, this can be solved by "lemmatization" in linguistics. For example, "happy" and "happiness" should be identified as the same word. In this project, we have solved this issue by utilizing the python `nltk` package’s `WordNetLemmatizer`.

2. Threshold self-adaptation is done by averaging sentence TF-IDF values, and identify the sentences retained by comparing to the relative TF-IDF mean.

3. Also, it is found that the code by Jain A. (2019) has wrongly calculated TF values since it divides the frequency by the length of sentence instead of the number of words.

5 Evaluation

We utilized `SequenceMatcher` in a Python library called `difflib` to evaluate the algorithm’s performance. In detail, we applied a data set (Grusky et al., 2018) including $995,040$ data points of news and given human-written summary. `SequenceMatcher` matches the output of the TF-IDF algorithm and the given summary each time the program runs and provides a float similarity parameter ranging from 0 to 1 (0 for totally different, 1 for identical). We then evaluate the algorithm’s performance by analyzing the raw parameters generated from each run.

We have defined two counts to evaluate and compare the performance of different selected values of threshold (percentage of retained content, the key parameter of the classical TF-IDF method) as well as parameterizing factor (by setting up the threshold as the averaged importance multiplied by a parameter, the key parameter of the self-adaptive TF-IDF method). One of the counts is called the Count of Optimal Value (COV). The COV of a certain parameter value calculates the number of data points regarding this value as the optimal one. The other count is called Total Similarity (TS). The TS of a certain parameter value is the sum of all related similarity parameters of this value in the data set.

For the classical TF-IDF method, the results of TS show that a value between $14\% - 16\%$ is optimal for the algorithm per se, while the results of COV indicate that a threshold of around 3% best matches the news. The two counts both indicate that 1.3 is the optimal value for the mean factor in the self-adaptive TF-IDF method. (See **Appendix** for the graph.)

The performance of the two methods is similar. The maximum TS of both methods is around $150,000$, which means neither of the methods outperforms the other.

6 Conclusion

The TF-IDF algorithm is efficient in terms of running time, and our evaluations also show its robustness. Also, the threshold of the self-adaptive TF-IDF method 1.3 matches with the article by Panchal (2019).

Appendix

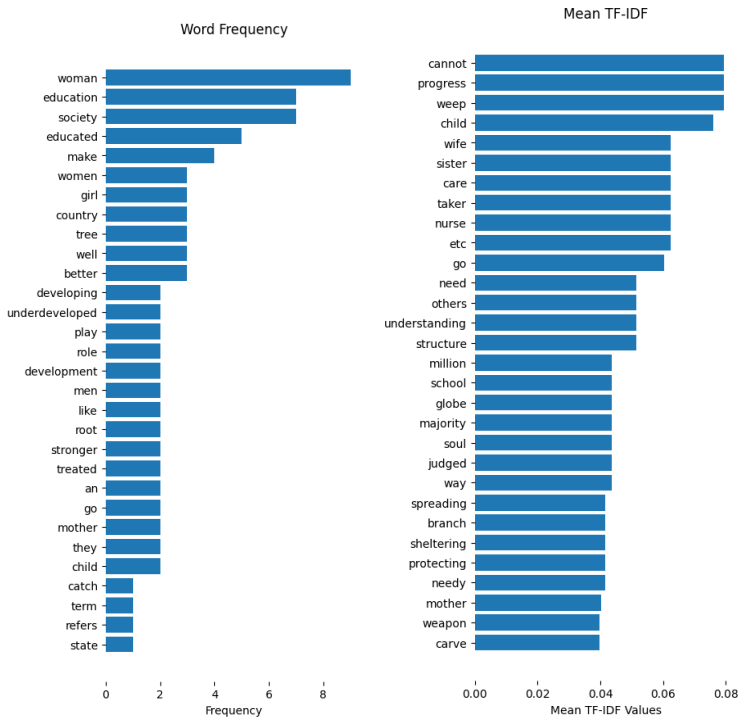


Figure 4: Left: the chart of word frequency after removing stopwords; Right: mean TF-IDF of each word (per appearance).

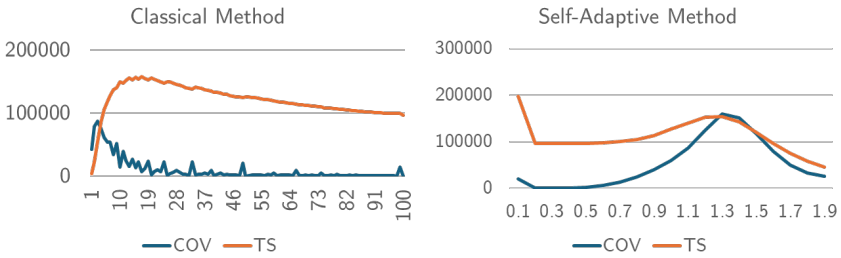


Figure 5: Left: Iterative testing of finding optimal threshold for the classical TF-IDF method; Right: Iterative testing of finding optimal parameterizing factor for the self-adaptive TF-IDF method.

References

[1] Robertson, S. (2004) Understanding Inverse Document Frequency: On Theoretical Arguments for IDF. In *Journal of Documentation* **60**(5):503–520.

[2] Sparck Jones, K. (1972) "A statistical interpretation of term specificity and its application in retrieval". In *Journal of Documentation*, Vol. 28, pp. 11–21.

[3] Manning, C.D., Raghavan, P. & Schütze, H. (2008) *Introduction to Information Retrieval*, pp. 117–120. Cambridge University Press.

[4] Jain, A. (2019) Automatic Extractive Text Summarization using TF-IDF. Accessed 18 November, 2023. `https://medium.com/voice-tech-podcast/automatic-extractive-text-summarization-using-tfidf-3fc9a7b26f5`. .

[5] Grusky, M., Naaman, M. & Artzi, Y. (2018) Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Volume 1 (Long Papers), pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

[6] Panchal, A. (2019) NLP — Text Summarization using NLTK: TF-IDF Algorithm. Accessed 21 November, 2023. `https://towardsdatascience.com/text-summarization-using-tf-idf-e64a0644ace3`.