

Examining extractive text summarization methods with CNN stories

Khushwant Kaswan, Jyoti Srivastava

Cite as: Kaswan, K., & Srivastava, J. (2023). Examining extractive text summarization methods with CNN stories. International Journal of Microsystems and IoT, 1(6), 402-409. <https://doi.org/10.5281/zenodo.10399814>




© 2023 The Author(s). Published by Indian Society for VLSI Education, Ranchi, India



Published online: 27 November 2023.



Submit your article to this journal: 




Article views: 



View related articles: 



View Crossmark data: 

DOI: <https://doi.org/10.5281/zenodo.10399814>



Examining extractive text summarization methods with CNN stories

Khushwant Kaswan, Jyoti Srivastava

Department of Computer Science, National Institute of Technology, Hamirpur, India

ABSTRACT

Because of exponential growth of content in digital format on the Internet, text summarization emerged as an important study area. This study presents an evaluation based on comparison of different extractive summarization techniques, namely Feature-based, Frequency-based, Graph-based and LSA based on the DeepMind CNN Stories dataset. The performance of these techniques was assessed using ROUGE metrics. The findings of our study indicate that the usage of the feature-based approach yielded superior results compared to the other techniques. This was evidenced by the attainment of the highest F1 scores in relation to ROUGE-1. Furthermore, the approach based on features generated summaries that were more comprehensible and informative compared to the other methodologies. The present investigation offers valuable perspectives on the efficacy of diverse extractive methods for summarization and emphasizes the capability of the feature-based approach to produce summaries of high quality.

KEYWORDS

Text Summarization; Extractive Text Summarization; DeepMind CNN; ROUGE; TF-IDF; LSA; TextRank; Feature Scoring

1. INTRODUCTION

Every year, the amount of content on the Internet nearly doubles [1]. The number of hosts in the DNS in July 2009 is 681,064,561 [2]. Text summarizing research began in the 1950s, yet no system exists today that can generate Gold Summaries (Professional Human Summaries). In general, text summaries are created in a step-by-step fashion. Text documents are first preprocessed. Preprocessing includes processes like stop word removal, stemming etc. In the next step sentences are scored on the basis of some criterion. Scored sentences are then selected to generate the final summary [3]. According to [4], text summarization is in high demand to address the ever-growing volume of text data available online to find relevant information more quickly.

Text summarization is not merely a convenience but a necessity in the contemporary information landscape. Its ability to distill large volumes of text into concise and meaningful summaries addresses fundamental challenges related to information overload, time constraints, and the need for efficient knowledge extraction, thereby making it an indispensable tool across diverse domains for Information Overload Mitigation, Enhanced Information Retrieval, Improved Comprehension, Multi-document Summarization for Aggregated Insights, Automation in Decision Support Systems, Accessibility Inclusivity and Personalization.

As the digital age continues to accelerate the production of textual content, summarization emerges as a crucial tool to

sift through, distill, and present information in a concise and digestible format.

Within the domain of text summarization, two primary approaches have gained prominence: extractive and abstractive summarization.

Extractive summarization focuses on the extraction of salient information directly from the source text. Unlike abstractive summarization, which involves generating new sentences to capture the essence of the document, extractive summarization identifies and selects existing sentences or phrases from the original text to compose the summary. This method aims to preserve the original wording and coherence of the source material while distilling its key content by Sentence Ranking Methods. These methods assign scores to sentences based on suitable criteria. The methods heavily relies on features derived directly from the source text. This could include statistical measures, linguistic features, or semantic analysis to identify sentences that capture the essence of the document. The sentences with the highest scores are then included in the summary. Extracted sentences are typically kept unchanged, ensuring that the summary reflects the language and nuances of the original document.

While extractive summarization has its advantages, it also faces challenges such as redundancy, coherence issues, and potential information loss. Striking the right balance between inclusion and exclusion of sentences poses a continual challenge in achieving optimal summarization results.

Summarization poses a challenge in terms of evaluation as it

necessitates a database containing text and their corresponding summaries that have been written or curated by humans. The definition of a good summary is an open-ended question[22]. These summaries, often referred to as "gold summaries", are challenging to obtain in real-world scenarios, and hence, research in this field tends to concentrate on news articles and scientific papers where they are most easily accessible. We have used DeepMind Q&A CNN/Daily Mail dataset[4].

The DeepMind Q&A CNN dataset contains 92,579 news articles and their respective summaries termed as highlights in form of separate paragraphs with @highlight tag.

The dataset was made available by New York University.

We parsed the dataset to create stories and their summaries in separated individual files. The separated stories and their summaries are then used to create a Pandas data frame which is treated as our raw corpus and needs cleaning.

Zero, one or two sentences, patterns that do not contribute much to the article and interview conversations are removed before summarization. The cleaned corpus contains 90,887 articles with their summaries.

The articles are cleaned up during the preprocessing phase before summarization by performing operations such as word and sentence level tokenization, lemmatization, stemming, Part of Speech tagging, removing stopword and punctuation. A thorough explanation of each of these preprocessing tasks is provided.

- *Sentence Tokenization*: The sentence-by-sentence breakdown of the article using sentence segmentation, which takes into account boundary constraints like a full stop or question mark.
- *Word Tokenization*: Tokenization is a process whereby phrases are broken down into their component words and special characters like spaces, commas, colons, semicolons, dashes, hyphens, closing brackets, quotations, exclamation marks, etc. are excluded.
- *Stop-Words Removal*: Stop words refer to frequently appearing words that don't significantly add to the document's meaning. The NLTK library can be utilized to iterate through all words and eliminate stop words.
- *Removing Punctuations*: Punctuation characters are obsolete symbols present in our corpus documents. We are going to remove these - `!'"#$%&()*+-./:;<=>@[\\]^_`{|}~\n`
- *Stemming and Lemmatization*: Words are reduced to their stems through the process of stemming, while lemmatization involves reducing words to their root form. Stem need not be a dictionary word while lemmatization reduces token to a root synonym and will be a dictionary word. The Porter-Stemmer from

the NLTK library was utilized for the purpose of stemming. For Lemmatization we have used NLTK library's WordNetLemmatizer.

- *POS Tagging* : This method classifies tokens/terms into the Part of Speech groups., such as nouns, pronouns, verbs, adverbs etc

The effectiveness of extractive summarization systems is often measured using metrics such as precision, recall, and F1 score.

The remaining sections are, Section 2 describes the scoring methods we have used for extractive summarization approaches in detail. Experimental results and Analysis is in section 3. Conclusion and Future Work in section 4.

2. METHODOLOGY

Methods based on Extractive Summarization include a series of steps:

- Preprocessing of text for summarization
- A numeric value i.e. score assigned to each of the sentences based on method used.
- Ranking of the sentences using the score either by sorting or other ranking algorithm. Higher ranked sentences are preferred over lower ranked sentences for summary.

Different techniques have been used to find that numeric value i.e. score for each sentence. These methods assign scores to sentences based on a suitable criteria. The methods heavily relies on features derived directly from the source text. This could include statistical measures, linguistic features, or semantic analysis to identify sentences that capture the essence of the document. The methods we have used are Frequency based approach, Graph based approach, Latent Semantic Analysis based approach and Feature based approach.

2.1 Frequency Based Approach

The fundamental method for summarizing articles is the frequency-based approach. The utilization of frequency analysis allows for the identification of relevant and non-relevant elements within a given article through the assessment of word significance. Typically, the topic conveyed in an article is represented by the words that occur most frequently.

Normal frequency based approach

Following the removal of stop words, J.N. Madhuri et al. [9] offered a method to construct an extracted summary by ranking sentences based on the frequency of their terms. Sentence score is calculated by adding frequency of word of the sentence in the whole article then normalizing using sentence word count. It's universally applicable, but cannot differentiate between sentences on a semantic level and also gives more score to lengthy sentences.

TF-IDF Method

The TF-IDF method mitigates the influence of frequently appearing words that possess a higher frequency in the corpus by associating them with their respective occurrence count in the set of documents. The TF-IDF methodology prioritizes uniqueness in generating the feature vector, thereby addressing the challenge of rarity of a given term

$$TF - IDF(t) = TF(t) * IDF(t)$$

$$TF(t) = \frac{Frequency(t)}{Total\ number\ of\ terms\ in\ the\ document}$$

$$IDF(t) = \log_e\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it}\right)$$

TF-IDF has different kinds of term weighting schemes[15]. Many summarizers [2, 3] utilize this technique since TF-IDF weights are simple and quick to calculate and also serve as effective indicators of the significance of words.

N-gram frequency based approach

We have modified the normal frequency based approach using n-gram where instead of using a single word we are using a sequence of n words. Sentence score is calculated by adding frequency of each n-gram item of the sentence in the whole article and then normalizing using sentence n-gram count.

$$Sentence\ Score = \frac{\sum Frequency(n - gram - item)}{No\ of\ n - gram - items\ in\ sentence}$$

Now we select the required number of sentences by sorting this score.

2.2 Graph Based Approach - TextRank

Focus for extractive text summarization is to rank the sentences based on their score. Here score can be calculated using the PageRank Algorithm[14] through the sentence_similarity_graph created between all the sentences of the article. Once the sentences have been ranked, the summary is generated by including the most significant sentences. This can be achieved through the application of either threshold values or selection of the top n sentences with respect to the scores.

The similarity matrix will be a square matrix consisting of the sentence similarity of a sentence with every other sentence in the document.

To calculate sentence similarity score between two sentences, we will be vectorising both the sentences and then calculating the cosine distance between both vectors as a similarity score[8].

To vectorize sentences, we are using two different methods. One involves normal vectorization using sentence length while other uses Glove Embeddings.

2.3 Semantic based approach - LSA

The concept of LSA - Latent Semantic Analysis was introduced by [20]. The fundamental concept underlying the utilization of

LSA for summarization is Words that tend to appear in similar contexts are likewise associated within the same singular unified space [21].

The Latent Semantic Analysis (LSA) technique is utilized to convert sentence vectors from a term space (non-orthogonal features) to a concept space with a lower dimensionality and an orthogonal basis. This is done by performing singular value decomposition of term sentence matrix A. The outcome of the process yields three matrices, namely U, V, and Σ :

$$A = U \Sigma V^T$$

where,

A is the input term sentence matrix (m×n);

U is words × extracted concepts (m×n);

Σ is diagonal descending matrix (n×n) and represents scaling values;

V^T is sentences × extracted concepts (n×n) [6].

The vectors that correspond to the terms and sentences in the concept space are respectively represented by the columns of matrix U and the rows of matrix V^T . By selecting the top k concepts based on their eigenvalues, we can obtain the optimal k-rank approximation of matrix A through the least squares method [7].

The score of a sentence, if V is the sentence vector and σ is the eigen-value of the ith concept in the concept space, is calculated as

$$Score(V) = \sqrt{\sum_i \sigma_i^2 v_i^2}$$

Steps followed during the process :

- For the preprocessed sentences calculate TF-IDF weights
- Get the sentence vectors V using the TF-IDF values.
- Applying SVD to the vectors obtained using TF-IDF weights
- Obtain the k highest singular values S utilizing the SVD.
- Utilize thresholding to eliminate singular values. This is a heuristic, and we can choose values according to our preference or take it as the mean of all values.
- To obtain the weights of sentences per topic, it is necessary to multiply each column of the term sentence matrix V by its corresponding squared singular value from matrix S.
- To obtain the salience scores for each sentence in a given document, the weights of the sentences across the topics should be added up first. Subsequently, the square root of the resulting score should be computed.

2.4 Feature based approach

The calculation of the sentence score involves performing a simple linear combination of the values of the features and weights associated with the feature.

$$Score = \sum w_i * F_i$$

We must be careful while minimizing the amount of sentences to ensure that the remaining sentences convey the vital information. So it is important that we include all crucial sentences while reducing the number of sentences. [10].

Features have been categorized into two kinds.

- *Frequency based Features* - There are various techniques for generating a document's summary, but the most straightforward method for summarizing articles is the frequency-based method [11].
- *Texture based Features* - To score all sentences in the article, textual features are used and further can be grouped as word and sentence levels. Then, the high-scoring sentences are chosen in order to produce a summary [12].

The order and structure of the sentences are maintained in the extractive summary. There are various methods for choosing sentences for the summary. It can be done either by deciding the length of the summary[13] or by thresholding wrt the scores. The sentences are arranged in descending order of scores and scores smaller than the threshold are not used in the summary while maintaining cohesiveness.

The text summary was created using the top 10 features that researchers have most frequently utilized over the last 10 years.

The many word level and sentence level features to score the sentences are covered in this section.

1. *Term Frequency F1*: The total frequency of the terms in each sentence was used to grade each sentence. Term repetition has a greater influence on the final score. This can occasionally result in an inaccurate summary.

$$F1 = \frac{\text{Frequency}(\text{term})}{\text{Number of terms present in the article}}$$

2. *Sentence Position F2*: In general, the beginning and last few sentences of a text contain more crucial information than the rest.

$$F2 = 1 - \frac{i-1}{N}$$

3. *Sentence Length F3*: Both extremely short and very long texts are filtered using this feature.

$$F3 = \frac{|S_i|}{\max|S|}$$

4. *Title Words Similarity F4*: Sentences containing title words are considered extremely significant compared to others as these sentence represent an aspect of the article and are the most relevant in a summary
 $F4 = \text{No of title words present in } S_i$

5. *Cue Words F5*: This feature assigns a score to sentences based on the occurrence of cue words like "As a Conclusion," "As a result," "Last of all". A cue word list must be used.
 $F5 = \text{No of cue words present in } S_i$

6. *Proper Nouns F6*: A proper noun denoted by an individual, location, or organization has higher significance to the article.
 $F6 = \frac{\text{No of Proper Nouns present in } S_i}{|S_i|}$

7. *Pronouns F7*: If pronouns (he, she, etc.) appear in the sentence, the entire sentence is given greater weight (the normalized frequency of pronouns).
 $F7 = \frac{\text{No of Pronouns present in } S_i}{|S_i|}$

8. *Numerical Values in Sentences F8*: The sentences that give numerical information, such as a date or transactions, are crucial and are included in the summary of the text.
 $F8 = \frac{\text{No of Numerical data present in } S_i}{|S_i|}$

9. *Thematic Features F9*: A list of the words for the primary domain are known as themed/thematic words. Thematic words are identified by selecting the most frequent words in the whole article.
 $F9 = \text{No of thematic words in } S_i$

10. *Word Co-occurrence F10*: There is a possibility that certain terms may co-occur within sentences. Assigning greater weight to co-occurring words can enhance the prominence of significant information in the final summary. This can be done by creating a similarity matrix for all the sentences and sentence score for a particular sentence can be assigned by normalizing the maximum similarity value of a sentence with the length of that sentence.
 $F10 = \frac{\text{Max}(\text{Similarity value of words in } S_i)}{|S_i|}$

Table. 1 Texture Based Features

Feature No	Category	Name
F ₁	Frequency Based	Term Frequency
F ₂	Sentence Level	Sentence Location
F ₃	Sentence Level	Sentence Length
F ₄	WordLevel	Title Words similarity
F ₅	WordLevel	Cue Words
F ₆	WordLevel	Proper Nouns
F ₇	WordLevel	Pronouns
F ₈	WordLevel	Numerical Values
F ₉	WordLevel	Thematic Features
F ₁₀	Word Level and Sentence Level	Word Co-occurrence

3. EVALUATION AND RESULTS

Precision, Recall and F-score are used for evaluation of these methods. These matrices assess how well the system's summaries or candidate summaries resemble golden summaries that are produced by humans or reference summaries. By calculating the amount of overlapping words, this comparison is made.

$$Precision = \frac{S_{reference} \cap S_{candidate}}{S_{candidate}}$$

$$Recall = \frac{S_{reference} \cap S_{candidate}}{S_{reference}}$$

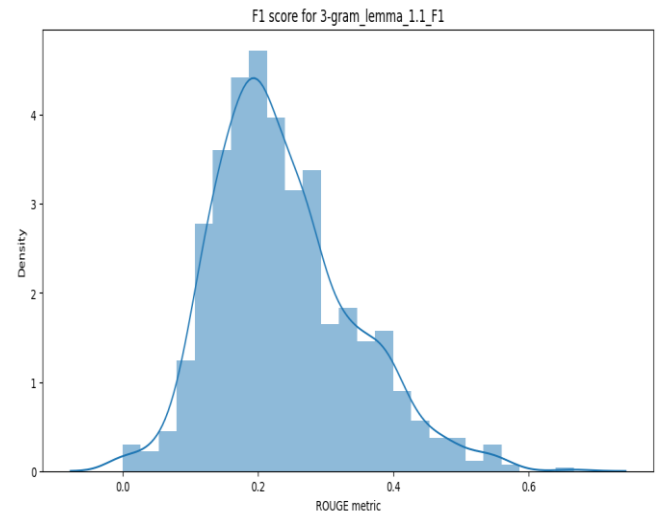
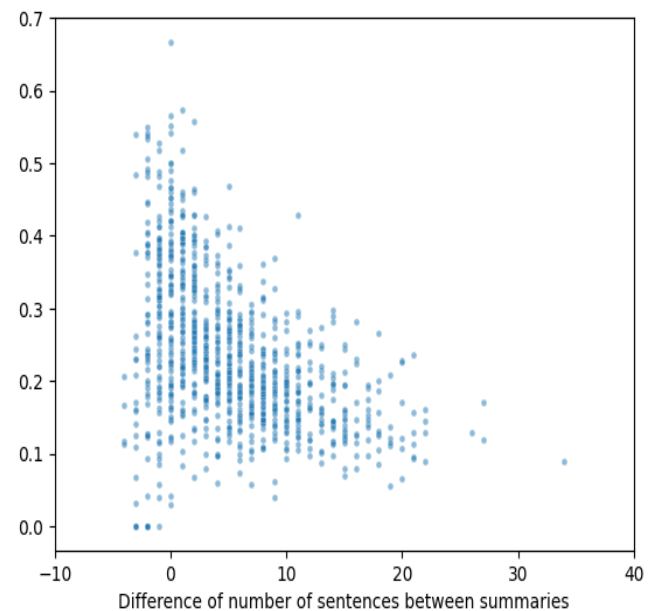
$$F - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

ROUGE- Recall-Oriented Understudy for Gisting Evaluation – Tool that is frequently utilized in the automated assessment of system generated summaries. The fundamental concept underlying ROUGE involves quantifying the degree of overlap between candidate and reference summaries, through the identification of shared units such as n-grams.

There exist multiple ROUGE metrics but we will be using ROUGE-N where N refers to degree of n-gram considered between candidate and reference summaries. We will be using ROUGE-1. This metric utilizes a uni-gram approach.

While some extractive summarization methods obtain high ROUGE scores, they all suffer from low readability[30].

		Mean				
		1	1.1	1.2	1.3	1.4
1-gram	stem	20.26	22.49	23.51	22.94	20.93
	lemma	20.31	22.46	23.51	22.79	21.02
2-gram	stem	20.84	23.13	23.48	21.27	18.64
	lemma	20.83	23.15	23.32	21.41	18.64
3-gram	stem	21.39	23.54	22.57	19.97	16.67
	lemma	21.43	23.69	22.54	19.74	16.54

Fig. 1 Results of N-gram Frequency Method**Fig. 2** F1 Score Graph for N-gram Frequency Method (x = F1_score, y = Density)**Fig. 3** N-gram Frequency Method (x = diff_number_of_sentences, y = F1_score)

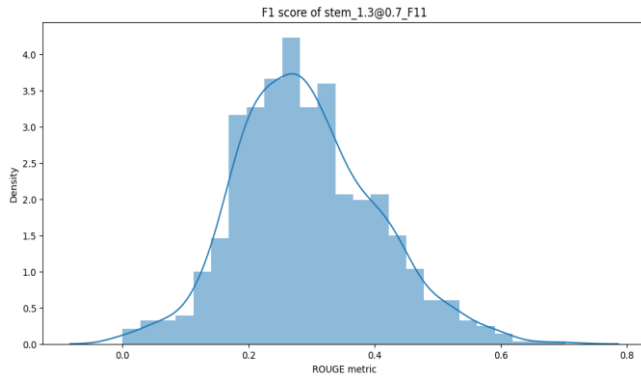


Fig. 4 F1 Score Graph for Feature Based Approach
(x = F1_score, y = Density)

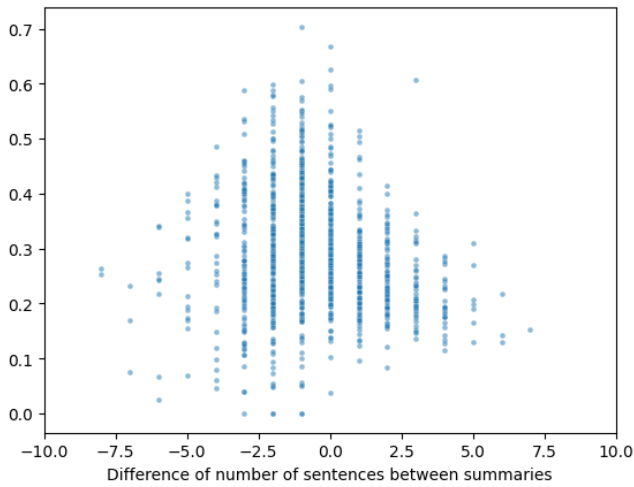


Fig. 5 Feature Based Approach
(x = diff_number_of_sentences, y = F1_score)

Table. 2 Extractive Text Summarization on 1000 articles

Approach	F1 Score
N-gram Frequency Based	23.69
TextRank - Normal	25.26
TextRank - Glove Embed.	25.76
LSA	18.56
Feature Based Approach	30.18

The frequency based approach is considered the simplest approach among extractive summarization techniques but based on the F-Score it has surpassed Latent Semantic Analysis (LSA) based approach[23].

Table. 3 F1 Score - Feature Based Text Summarization on random gold summaries

Document	F1 Score
Doc1	43.03
Doc2	25.73
Doc3	37.58
Doc4	51.48
Doc5	29.72
Doc6	26.1
Doc7	26.49

Table. 4 Comparison with different extractive and abstractive summarization techniques on the dataset

	Rouge-1
Frequency-based [23]	19%
GPT-2 no hint [25]	21.58%
N-gram Frequency Based*	23.69%
SD-KMeans [23]	24%
Random-3 [25]	28.78%
GPT-2 TL;DR [25]	29.34%
Hybrid MemNet [24]	29.9%
Feature Based Approach*	30.18%
REFRESH [24]	30.4%
ITS [24]	30.8%
Seq2Seq [26]	35.5%
SOTA [29]	41.22%
HSSAS [28]	42%

Comparison with standard publications results

- As per results provided in [23], the normal frequency based approach yields an F score of about 19% while our N-gram based frequency approach yields about 24%

- As per results provided in [23], the LSA based approach yields an F score of about 16% while our LSA implementation approach yields about 18%
- As per results provided in [23], SD-KMeans yield about 24% which is in comparison to our Graph based approaches i.e. 25% but less than Feature based i.e. 30%
- As per results provided in [24], Cheng et.al'16 yields 28.4% and Hybrid MemNet yields 29.9% and As per [25], GPT2 produces a score of 29.34% and Random-3 produces 28.78% while our best F1 score is about 30% for Feature Based Approach.
- ChatGPT-3 yields about 23.53% [23] which is less than our scores.
- As per results provided in [24], both the REFRESH model gives 30.4 % and ITS model yields 30.8% and the scores are marginally higher to the Feature Based Approach 30.18%.
- The LEAD-3 baseline is a commonly used baseline in news summarization that extracts the first three sentences from an article[29]. As per [24] full-length F1 variant Lead-3 results in 29.1% while [25] shows 40.38% while our score is 30.18%.
- Our score of 30% is far less compared to the SOTA model 41.22% [25], Sequence-to-Sequence RNNs [26] 35.5%, FineTuned BERT [27] 43% and HSSAS [28] 42%.

4. CONCLUSION AND FUTURE WORK

In Table 2, we can see how several extractive summarization tools performed on the dataset and comparison of our results with standard results in Table 4. Our **N-gram based frequency approach** has scored better compared to normal frequency approach and LSA based approaches. **Feature Based Approach** performed better than many extractive summarization techniques but lacks when compared to advanced abstractive techniques.

One of the reasons for the performance drop is because of few bogus articles and lack of semantics, but it can be raised again if the dataset is cleaned up more deeply. In future, utilization of enhanced preprocessing techniques and a deeply cleaned corpus of contextualized data may produce more precise results.

Also in Feature Based Approach, we have not considered weights of individual features and will attempt to improve our outcomes by associating weights with the features we have utilized thus far. Genetic algorithms can be utilized to determine accurate weights for the features.

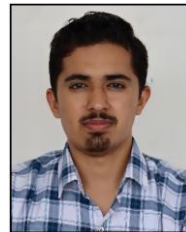
In addition, we will attempt to enhance the present version of the Summarizer by incorporating some semantic elements and Word Net. Also, we can adopt word embeddings to include semantics in normal statistical algorithms.

REFERENCES

- [1] D. R. Radev and W. Fan. (2000). "Automatic summarization of search engine hit lists", Proceedings of the ACL-2000 workshop on recent advances in natural language processing and information retrieval, Hong Kong, pp. 99-109. <https://doi.org/10.3115/1117755.1117768>
- [2] ISC (2009). "ISC Internet Domain Survey", December, 2009, <http://ftp.isc.org/www/survey/reports/current/>
- [3] Yogesh Kumar Meena and Dinesh Gopalani. (2014). Analysis of Sentence Scoring Methods for Automatic Text Summarization. In Proceedings of the 2014 International Conference on Information and Communication Technology for Competitive Strategies (ICTCS '14). Association for Computing Machinery <https://doi.org/10.1145/2677855.2677908>
- [4] Karl Moritz Hermann and Tomáš Kočiský and Edward Grefenstette and Lasse Espeholt and Will Kay and Mustafa Suleyman and Phil Blunsom. (2015). "Teaching Machines to Read and Comprehend", Advances in neural information processing systems. <https://doi.org/10.48550/arXiv.1506.03340>
- [5] S. L. Hou, X. K. Huang, C. Q. Fei et al. (2021). "A survey of text summarization approaches based on deep learning," Journal of Computer Science and Technology, vol. 36, no. 3, pp. 633-663. <http://dx.doi.org/10.1007/s11390-020-0207-x>
- [6] M. G. Ozsoy, F. N. Alpaslan, and I. Cicekli. (2011). "Text summarization using latent semantic analysis," Journal of Information Science, vol. 37, no. 4, pp. 405-417. <https://doi.org/10.1177/0165551511408848>
- [7] Shrabanti Mandal, & Girish Kumar Singh. (2020). LSA Based Text Summarization. International Journal of Recent Technology and Engineering (IJRTE), 9(2), 150-156. <https://doi.org/10.35940/ijrte.B3288.079220>
- [8] Mallick, Chirantana, Ajit Kumar Das, Madhurima Dutta, Asit Kumar Das, and Apurba Sarkar. (2019). "Graph-based text summarization using modified TextRank." In Soft computing in data analytics, pp. 137-146. Springer, Singapore, 2019. http://dx.doi.org/10.1007/978-981-13-0514-6_14
- [9] J. N. Madhuri and R. Ganesh Kumar. (2019). "Extractive Text Summarization Using Sentence Ranking," 2019 International Conference on Data Science and Communication (IconDSC), Bangalore, India, 2019, pp.1- 3. <https://doi.org/10.1109/IconDSC.2019.8817040>
- [10] Sunil Dhankhar & Mukesh Kumar Gupta. (2022). A statistically based sentence scoring method using mathematical combination for extractive Hindi text summarization, Journal of Interdisciplinary Mathematics, 25:3, 773-790, <https://doi.org/10.1080/09720502.2021.2015096>
- [11] Saeed, M.Y., Awais, M., Talib, R., Younas, M.(2020). "Unstructured Text Documents Summarization with Multi-Stage Clustering," IEEE Access, 8, 212838-212854. <http://dx.doi.org/10.1109/ACCESS.2020.3040506>
- [12] D. Patel, S. Shah, and H. Chhinkaniwala. (2019). "Fuzzy logic based multi document summarization with improved sentence scoring and redundancy removal technique," Expert Systems with Applications, 134:167-177. <https://doi.org/10.1016/j.eswa.2019.05.045>
- [13] Liu W, Gao Y, Li J, Yang Y. (2021). "A Combined Extractive with Abstractive Model for Summarization," IEEE Access 43970-43980. <https://ieeexplore.ieee.org/iel7/6287639/9312710/09380377.pdf>
- [14] Mihalcea R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. In: Proceedings of the ACL 2004 on Interactive Poster and

- Demonstration Sessions, pp. 20. Association for Computational Linguistics <https://doi.org/10.3115/1219044.1219064>
- [15] Gerard Salton and Christopher Buckley. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management* 24, 5 (1988), 513–523. [https://doi.org/10.1016/0306-4573\(88\)90021-0](https://doi.org/10.1016/0306-4573(88)90021-0)
- [16] Rasim M Alguliev, Ramiz M Aliguliyev, Makrufa S Hajirahimova, and Chingiz A Mehdiyev. (2011). MCMR: Maximum coverage and minimum redundant text summarization model. *Expert Systems with Applications* <http://dx.doi.org/10.1016/j.eswa.2011.05.033>
- [17] Goularte F. B, Nassar S. M, Fileto R, Saggion H. (2019). A text summarization method based on fuzzy rules and applicable to automated assessment. *Expert Syst. Appl.* 2019, 115, 264–275. <https://doi.org/10.1016/j.eswa.2018.07.047>
- [18] Rasim M Alguliev, Ramiz M Aliguliyev, and Nijat R Isazade. (2013). Multiple documents summarization based on evolutionary optimization algorithm. *Expert Systems with Applications* 40, 5 (2013), 1675–1689. <https://doi.org/10.1016/j.eswa.2012.09.014>
- [19] Mehdi Allahyari, Seyedamin Pouriye, Mehdi Assefi, Saeid Safaei, Elizabeth D Trippe, Juan B Gutierrez, and Krys Kochut. (2017). Text summarization techniques: a brief survey. *arXiv preprint arXiv:1707.02268*. <https://doi.org/10.48550/arXiv.1707.02268>
- [20] Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. (1990). Indexing by latent semantic analysis. *JASIS* 41,6 (1990), 391–407. [https://doi.org/10.1002/\(SICI\)1097-4571\(199009\)41:6%3C391::AID-ASI1%3E3.0.CO;2-9](https://doi.org/10.1002/(SICI)1097-4571(199009)41:6%3C391::AID-ASI1%3E3.0.CO;2-9)
- [21] Kaiz Merchant and Yash Pande. (2018). Nlp based latent semantic analysis for legal text summarization. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pages 1803–1807. IEEE. <https://doi.org/10.1109/ICACCI.2018.8554831>
- [22] Horacio Saggion and Thierry Poibeau. (2013). Automatic text summarization: Past, present and future. In *Multi-source, multilingual information extraction and summarization*, pages 3–21. Springer http://dx.doi.org/10.1007/978-3-642-28569-1_1
- [23] Nam Ho Koh, Arshdeep Singh, Joe Plata. (2022). SummIt!: An in-depth analysis of 4 automatic text summarization methods. *EECS 595 Fall 2022, University of Michigan* https://sled.eecs.umich.edu/media/eecs595_fa22/16_Koh_Singh_Plata.pdf
- [24] Xiuying Chen and Shen Gao and Chongyang Tao and Yan Song and Dongyan Zhao and Rui Yan. (2018). Iterative Document Representation Learning Towards Summarization with Polishing. *arXiv:1809.10324* <https://doi.org/10.48550/arXiv.1809.10324>
- [25] Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei and Ilya Sutskever. (2019). “Language Models are Unsupervised Multitask Learners.” <https://api.semanticscholar.org/CorpusID:160025533>
- [26] Nallapati, Ramesh, Bowen Zhou, Caglar Gulcehre, and Bing Xiang. (2016). “Abstractive text summarization using sequence-to-sequence RNN and beyond.” *arXiv preprint arXiv:1602.06023*. <https://doi.org/10.18653/v1/K16-1028>
- [27] Liu, Yang. (2019). “Fine-tune BERT for Extractive Summarization.” *ArXiv abs/1903.10318*. <https://doi.org/10.48550/arXiv.1903.10318>
- [28] Al-Sabahi, Kamal, Zhang Zuping, and Mohammed Nadher. (2018). “A hierarchical structured self-attentive model for extractive document summarization (HSSAS).” *IEEE Access* 6: 24205–24212. <https://doi.org/10.1109/ACCESS.2018.2829199>
- [29] Gehrman, S., Deng, Y., and Rush, A. M. (2018). Bottom-up abstractive summarization. *arXiv:1808.10792*. <https://doi.org/10.48550/arXiv.1808.10792>
- [30] Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. (2018). A unified model for extractive and abstractive summarization using inconsistency loss. *arXiv:1805.06266* <https://doi.org/10.48550/arXiv.1805.06266>

AUTHORS



Khushwant Kaswan received his B.Tech degree from National Institute of Technology, Hamirpur, Himachal Pradesh, India in Computer Science and Engineering in 2023. His areas of interest are Natural Language Processing, Artificial Intelligence & Machine Learning. Email: khushwantk567@gmail.com



Jyoti Srivastava is working as an Assistant Professor in National Institute of Technology Hamirpur, Himachal Pradesh, India. She received her Ph.D. in 2018 (TCS Research Scholar) from Indian Institute of Information Technology (IIIT), Allahabad, Uttar Pradesh, India. She did her M. Tech in 2009 (GATE Scholarship) in Information Technology with specialization in Human Computer Interaction from IIIT Allahabad with 10 CGPA. She has published several research papers in reputed international conferences like IALP, CICLing, JapTAL and reputed journals (SCI & SCOPUS) like ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), Journal of Intelligent & Fuzzy Systems, etc. Her current research areas of interest include Natural Language Processing, Machine Translation, Artificial Intelligence. She is a reviewer of several reputed SCI journals. Corresponding author Email: jyoti.s@nith.ac.in