Extractive Text-Image Summarisation in Hindi

Pratik Savla[a], Sahil Jaiswal[b], Dr. G. Manju[c]

[a,b,c] Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur – 603 203, India

[a] pm5808@srmist.edu.in, [b] sr4175@srmist.edu.in, [c] manju.g@ktr.srmuniv.ac.in

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 4 June 2021

Abstract: Today's world has skyrocketed by the gathering and dissemination of huge amounts of data. A lot of this data is in text form which makes it very difficult to store and process. Hindi is the national language of India. Dataset of text-image summarization is not readily available for Hindi language and hence we created a dataset of 40558 news articles with images for the task and created extractive summaries for them. We did the evaluation using ROGUE and BELU metrics.

Keywords: dataset, text-image summarisation, Hindi, extractive summarisation

1. Introduction

In this modern world data is generated constantly in huge volume. This data can be in the form of audio, video, text, images and sensor data. As humans we use text data extensively every day. To understand this text data more quickly and generate insights summarizing it will help a lot. Text rundown is generally utilized by a few sites and applications to make news channel and article synopses. We incline toward short outlines with every one of the significant focuses over perusing an entire report and summing up it ourselves. Outline is a procedure to abbreviate long messages to such an extent that the synopsis has every one of the significant places of the real record. Ways to deal with Automatic Summarisation:Extraction-based Summarization and Abstraction-based Summarization.Extractive synopsis points to make a summary by choosing a subset of the sentences in the information text that expands the inclusion of significant content while limiting excess. Conversely, abstractive rundown means to make a theoretical portrayal of the information text and utilize common language age techniques to produce a synopsis. In contrast with extractive outlines, abstractive synopses are more difficult to produce, yet are seemingly a superior estimation of human outlines as they may contain articulations that don't exist in the first content (Cohn and Lapata 2008).

Extractive summarization aims to create a summary byselecting a subset of the sentences in the input text thatmaximizes the coverage of important content whileminimizing redundancy. In contrast, abstractivesummarization aims to create an abstract representation of theinput text and use natural language generation techniques togenerate a summary. In comparison to extractive summaries,abstractive summaries are more challenging to produce, butare arguably a better approximation of human summaries asthey may contain expressions that do not exist in the originaltext (Cohn and Lapata 2008). Table 1 shows an inputdocument and the corresponding human-generated abstractivesummary.

The focal point of text synopsis research has shown a steady move from extractive procedures to abstractive techniques lately, owing to some degree to huge advances in the advancement of neural techniques. Initially created for machine interpretation, neural techniques have ostensibly revolutionized the manner in which abstractive synopsis research is directed, making new, energizing freedoms for summarisation and age specialist.

2. Literature Survey

Early ways to deal with summarisation include: (1) sentence pressure (Cohn and Lapata 2009), which means to make a linguistic outline of a given sentence; (2) sentence combination (Barzilay and McKeown 2005; Filippova and Strube 2008), which includes utilizing base up nearby multisequence arrangement to distinguish phrases passing on comparative data and factual age to join normal expressions into a sentence; and (3) sentence correction (Tanaka et al. 2009), which produces sentences not found in the info and integrates data across sentences.

The previously mentioned approaches offer little improvement over extractive techniques, notwithstanding. This inspires the advancement of a completely abstractive methodology, which normally contains three subtasks acted in a pipeline design: data extraction, content choice, and surface acknowledgment.

Data extraction plans to remove significant data from the info text. Numerous abstractive summarizers centre around extricating phrasal-level data, for example, thing phrases (NPs) and action word phrases (VPs) along with their relevant data (Genest and Lapalme 2012; Bing et al. 2015). Mehdad et al. (2014) utilize query based extraction, which plans to extricate significant substance utilizing consequently produced inquiries and channel substance that have a low likelihood of being remembered for a rundown. Genest and Lapalme (2012) separate Information Items (INITs), which they characterize as the littlest component of lucid data in a sentence. Solidly, an INIT is characterized as a dated and found subject-action word object triple. Some space explicit summarizers utilize information on the class, theme, or area of the contribution to direct the sort of data to be removed (Wang and Cardie 2013). Review from the past area that in guided summarisation, the angles for a classification (e.g., Attacks) are given. Therefore, extraction rules can be planned dependent on deliberation patterns explicit to a specific class to extricate the ideal data. For instance, a murdering composition necessitates that the executioner, the action word that triggers the slaughtering occasion, and the casualty be extricated. Sometimes, nonetheless, the information report covers different points, which make manual pre-labelling of the record troublesome. For instance, in gathering record rundown, a few points might be referenced during the gathering (Oya et al. 2014), in which case subject division can be applied to distinguish the themes.

In diagram based techniques, charts are utilized to execute the previously mentioned three abstractive rundown subtasks. Diagrams are picked as a result of their expressiveness: they encourage the extraction of the ideas in an information archive as well as the conceivably intricate and dynamic relations between them (Greenbacker 2011). For instance, occasion semantic connection organizations (ESLNs) have been utilized for joint data extraction and substance determination (Li et al. 2016). Given an info text, an ESLN can be developed to give a theoretical portrayal of the content. In particular, every hub compares to an occasion referenced in the info text, where an occasion is made out of an occasion trigger/activity and its contentions. An edge between two hubs encodes the semantic connection between the comparing occasions. After network development, ILP can be applied to this organization to perform data extraction and substance choice (i.e., choosing a subset of hubs for creating the synopsis), utilizing requirements like Bing et al's. (2015) (e.g., the length limitations) just as imperatives characterized on the semantic relations (e.g., the hubs ought to be picked with the end goal that the subsequent chart stays associated).

3. Prposed Work

Datasets

The summarisation data is taken from Dainik Bhaskar Hindi news website. Dainik Bhaskar website link is https://www.bhaskar.com/. Articles for different categories (sports, national, international, entertainment, etc) withimages present in the article are extracted, total of 40558 article summary pair are generated, as shown in the Table 1. Gold summariesare generated using TextRank algorithm under human supervision. Each summary is of 5% of original text.

Table- I Categories of Articles

Categories	***Number of Articles***
Business	5068
Coronavirus	5197
Entertainment	7194
International	3172
National	7939
Sports	6034
Technology	3387
Utility	2567
Total	**40558**

The data was pre-processed after scrapping in the following ways:

- Articles which did not have images in it were rejected and not taken into consideration.
- In some of the articles, videos and non-textual descriptions were present which were ignored.
- All the links to other pages and news websites were removed from the articles.

Dataset for training the image feature extraction model is taken from Flickr8K Image Captioning dataset. It contains 8000 images with 5 captions for each image in English Language. We used Google translate api to translate the English captions into Hindi language for our use.

Feature Extraction

Image Features: The best way to extract features from images is to use convolution neural networks. We train the image captioning model using the data described previously to aid us in extracting the summaries. Then we propose to used the word embeddings generated by the captioning model in combination with the article embeddings and use the cosine similatiry technique to get the similarity score of article sentences with the images present in the articles. This is then used with other text features to rank the sentences and get the extractive text summay.

Text Features:

*Words occuring in heading of the article:*Weightage is given to words happening in title of the article when contrasted with different words, as those are significant. All the above loads are determined and standardized to a size of 0-1.

Length of the sentence: The size of sentence fluctuates a great deal inside the article, and we calculate the number of words and characters present in it. We then normalize it by dividing the length with total length of the article and use this number as of of the features for the sentence.

Position of the sentence inside article and in sections is also considered. We use similar method to calculation of the length of the sentence. The incentive for sentence position removed is standardized to take on qualities somewhere in the range of 0 and 1.

*Number of the verbspresent in a sentence:*We calculate the occurences of the verbs which indicate that the sentence is complete and may not rely of neighbouring sentences whichcan be the potential candidate for the extractive summary.

*Similarity of the sentence to the headline:*We calculate the cosine similatiry of the sentence vector with the headline embeddings and the feature is determined and mulled over.

Cosine similarity of a sentence: We consider different sentence embeddings and their comparability with one another. We process a likeness score of sentences in the accompanying manner: The addition of cosinesimilarity score with each and every sentence embedding in the report is thought of and normalized to give rank to every sentence.

Sentence Cohesion: This component is acquired for all the sentences in this way: first, we figure the embedding vector addressing the mean of the record, which is the number juggling normal and then comparing coordinate estimations of the multitude of sentences of the report; at that point we register the similitude between the centroid and Each sentence, getting the crude estimation of this element for each sentence. The standardized worth in the reach [0, 1] for s is acquired by processing the proportion of the Crude component esteem over the biggest crude element esteem among all sentences in the archive.

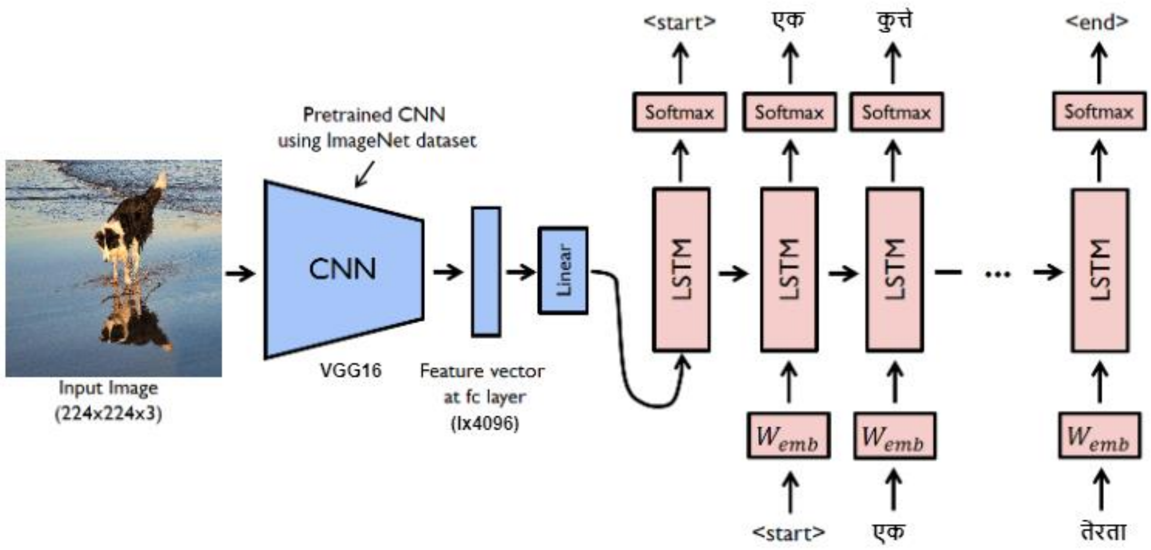

Fig- 1 Image Feature Extraction model architecture.

4. Implementation

Image Feature Extraction

We start by training the image captioning model which will give us the image features in the form of sentences to use in the extractive summary generation. We have divide the Flickr8k image captioning dataset into 3 parts: 5000 images in training set, 1000 images in dev set and 1000 images in testing set. We use a python script to get the Hindi version of the captions using google translate library. We have 5 Hindi captions for every image. Next we use two image models VGG16 and InceptionV3 to get the image encoding for the language model. Among all the different language models we trained, best results were achieved with combination of VGG16 for image encoding and 2-layer LSTM decoder model for generating the image caption. As shown in the Figure 1. We trained the decoder model on Nvidia GTX 1050 graphics card for 6 hours for 20 epochs. We achieved BLEU-1 score of 0.53836 and BELU-2 score of 0.347383. All the other result's scores and losses are mentioned in the results section.

Text Feature Extraction

We consider the seven features discussed in the proposed work section, which are occurrence in heading of articles, sentencelength,sentence position, presence of the verb in a sentence embedding, cosine similarity to the title of news article, cosine similarityof a sentence andsentence-to-centroidcohesion. We use word embeddings of the article sentences and the headline to determine the features and NLTK parts of speech tagger for checking the presence of verb. We convert everything into matrix and vectors for quick dot product operations over sentences. We combine all the feature scores by multiplying them with individual feature weights and then sum them for the final ranking of sentences using equation (1).

$$r_i = Sa_jS_j$$

Where r_i is score of i^{th} sentence, a_j is weight of the score S_j.

Post-processing

We get the top-n sentences from the original article from the ranking and use them as the final most important sentences in the news article.

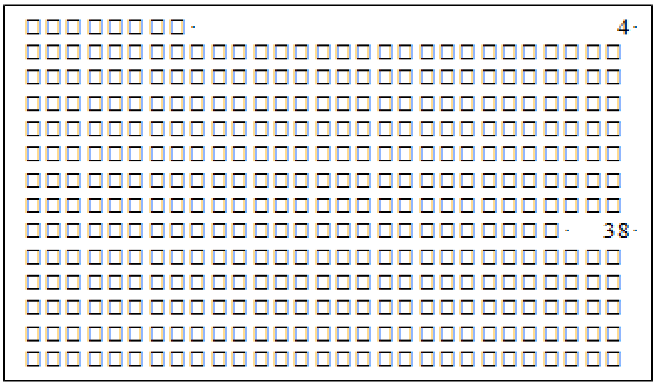

Fig-2 Summary generated by the model for an article from Dainik Bhaskar. URL: https://www.bhaskar.com/sports/news/football-league-is-the-first-to-start-because-its-turnover-is-137-of-spains-gdp-it-creates-1-lakh-85-thousand-jobs-127397734.html

5. Results discussion

Two types of evaluation methods are typically used to evaluate machine-produced summaries: manual evaluation and automatic evaluation.

In manual evaluation, human judges are asked to choose the best summary among several candidates by manually scoring each one along multiple dimensions of quality such as accuracy, clarity, and completeness (Greenbacker 2011). However, as manual evaluation is time-consuming and is particularly inefficient for large-scale evaluations, there have been a lot of attempts to develop automatic evaluation methods. For this reason, several automatic evaluation metrics have been developed. The widely-used metrics include (1) BLEU (Papineni et al. 2002), which was originally developed to evaluate machine translation systems; (2) METEOR (Denkowski and Lavie 2014), which addresses BLEU's weakness when applied to low-resource languages and has a better correlation with human judgment at the sentence/segment level than BLEU; (3) Pyramid (Nenkova et al. 2007), a wellknown method for evaluating content selection in summarisation; and (4) ROUGE (Lin 2004), a recall-based evaluation metric for summarization. Being one of the most popular metrics, ROUGE has several commonly used variants, such as ROUGE-N, which computes the n-gram recall between a candidate summary and a reference summary; ROUGE-SU, which uses skip-bigrams and unigrams to measure recall; 9815and ROUGE-L (Longest Common Subsequence), which requires in-sequence but not consecutive matches that reflect sentence-level word order n-grams.

Table-II Image Decoder BELU Scores

VGG16 image encoder				
Neurons in LSTM Layer	***B1***	***B2***	***B3***	***B4***
256	0.4669	0.288	0.184	0.083
128	0.538	0.347	0.2095	0.091
64	0.274	0.1456	0.091	0.036
ResNet-50 image encoder				
Neurons in LSTM Layer	***B1***	***B2***	***B3***	***B4***
256	0.314	0.1925	0.078	0.043
128	0.413	0.278	0.201	0.092
64	0.1717	0.085	0.06	0.024

$$F_1 score = \frac{2(precision * recall)}{(precision + recall)} \qquad (2)$$

By using the best image decoder with VGG16 encoder model we were able to get the ROGUE-N scores on the manually extracted gold summaries in Table III on various number of summary sentences. ROUGE-N (Recall-Oriented Understudy for Gisting Evaluation), a simple n-gram recall calculated between the set of summaries used as the reference and the candidate summary to evaluate, is a well-known technique for the same.

Table-III ROGUE-n Scores By Number Of Sentences

Number of Sentences	***Rogue-1***	***Rogue-2***	***Rogue-l***
3	0.460526	0.373333	0.486956
4	0.796019	0.753768	0.684210
5	0.727999	0.629032	0.584269
6	0.677852	0.540540	0.561576

Example in Fig. 2 is an article with many numbers, and even the human interpretation of the text is different. Summaries which are full of factoids, are difficult to interpret and depends on the readers perspective. The total number mentioned along with the ratio described yields result from one sentence to another. While a human reads the summary, the calculation, and understanding of other domains help, which is not the case of an algorithmic summary.

6. Conclusion

Our text-image summariser proposes feature extraction technique to generate summaries for Hindi language. We have achieved good results in comparison to other methods for the summaries we manually extracted for the summary evaluation dataset. Work done in this research has significant contribution to provide text-image dataset of 40558 News articles in Hindi. This dataset can be further used for training of other text-image problems as well. For evaluation purposes of extractive summarisation, we have manually annotated and written summary of these articles.

During our journey to build a text-image summariser for Hindi, we realized that there is much work that can be done in this domain, to further work on our results and to carry on the research in different domains.We computed the results on all the algorithms on manually extracted dataset of 40558 articles summary pair. Using these results, we can use it on various NLP applications like Sentiment Analysis, Question Answering system, cross-domain summarisation, and so on.

References

1. . Abe. A movement theory of anaphora, volume 120. Walter de Gruyter GmbH & Co KG, 2014.

2. J. Anitha, P. Prasad Reddy, and M. Prasad Babu. An approach for summarizing hindi text through a hybrid fuzzy neural network algorithm. Journal of Information & Knowledge Management, 13(04):1450036, 2014.
3. S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In Lrec, volume 10, pages 2200–2204, 2010.
4. P. B. Baxendale. Machine-made index for technical literature—an experiment. IBM Journal of Research and Development, 2(4):354–361, 1958.
5. P. Bhattacharyya. Indowordnet. In The WordNet in Indian Languages, pages 1–18. Springer, 2017.
6. R. Brandow, K. Mitze, and L. F. Rau. Automatic condensation of electronic publications by sentence selection. Information Processing & Management, 31(5):675–685, 1995.
7. G. Carenini and J. C. K. Cheung. Extractive vs. nlg-based abstractive summarization of evaluative text: The effect of corpus controversiality. In Proceedings of the Fifth International Natural Language Generation Conference, pages 33–41. Association for Computational Linguistics, 2008.
8. T.-M. Chang, W.-F. Hsiao, et al. A hybrid approach to automatic text summarization. In 2008 8th IEEE International Conference on Computer and Information Technology, pages 65–70. IEEE, 2008.
9. D. M. COE. A comparative study of hindi text summarization techniques: Genetic algorithm and neural network. 2015.
10. J. M. Conroy and D. P. O'leary. Text summarization via hidden markov models. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pages 406–407. ACM, 2001.
11. V. Dalal and L. Malik. Semantic graph based automatic text summarization for hindi documents using particle swarm optimization. In International Conference on Information and Communication Technology for Intelligent Systems, pages 284–289. Springer, 2017.
12. H. P. Edmundson. New methods in automatic extracting. Journal of the ACM (JACM), 16(2):264– 285, 1969.
13. V. R. Embar, S. R. Deshpande, A. Vaishnavi, V. Jain, and J. S. Kallimani. saramsha-a kannada abstractive summarizer. In Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on, pages 540–544. IEEE, 2013.
14. D. Goldhahn, T. Eckart, and U. Quasthoff. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In LREC, volume 29, pages 31–43, 2012.
15. Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pages 19–25. ACM, 2001.
16. E. Hovy and C.-Y. Lin. Automated text summarization and the summarist system. In Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998, pages 197–214. Association for Computational Linguistics, 1998.
17. S. Jha, D. Narayan, P. Pande, and P. Bhattacharyya. A wordnet for hindi. In International Workshop on Lexical Resources in Natural Language Processing, Hyderabad, India, 2001.
18. D. Kaur and R. Kaur. Automatic summarization of text documents written in hindi language. 2014.
    A. Khan. A review on abstractive summarization methods. 59:64–72, 01 2014.
19. F. Koto, S. Sakti, G. Neubig, T. Toda, M. Adriani, and S. Nakamura. The use of semantic and acoustic features for open-domain ted talk summarization. In Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific, pages 1–4. IEEE, 2014.
20. K. V. Kumar and D. Yadav. An improvised extractive approach to hindi text summarization. In Information Systems Design and Intelligent Applications, pages 291–300. Springer, 2015.
21. J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. In Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, pages 68–73. ACM, 1995.
22. C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. Text Summarization Branches Out, 2004.
23. C.-Y. Lin and E. Hovy. Identifying topics by position. In Proceedings of the fifth conference on Applied natural language processing, pages 283–290. Association for Computational Linguistics, 1997.
24. N. Loukachevitch and A. Alekseev. Summarizing news clusters on the basis of thematic chains. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014), 2014.
25. H. P. Luhn. The automatic creation of literature abstracts. IBM Journal of research and development, 2(2):159–165, 1958.

26. F. Moawad and M. Aref. Semantic graph reduction approach for abstractive text summarization. In Computer Engineering & Systems (ICCES), 2012 Seventh International Conference on, pages 132–138. IEEE, 2012.
27. D. Narayan, D. Chakrabarti, P. Pande, and P. Bhattacharyya. An experience in building the indo wordnet-a wordnet for hindi. In First International Conference on Global WordNet, Mysore, India, 2002.
28. M. Osborne. Using maximum entropy for sentence extraction. In Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4, pages 1–8. Association for Computational Linguistics, 2002.
29. L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank algorithm: Bringing order to the web. In Proceedings of the International Conference on the World Wide Web, 1998.
30. V. Rai, S. Vijay, and D. Misra. Linguistic approach based transfer learning for sentiment classification in hindi. In Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017), pages 373–382, Kolkata, India, December 2017. NLP Association of India.
31. V. Rai, S. Vijay, and D. M. Sharma. A karaka based approach to cross lingual sentiment analysis.
32. M. Subramaniam and V. Dalal. Test model for rich semantic graph representation for hindi text using abstractive method. International Research Journal of Engineering and Technology (IRJET), 2(2), 2015.
33. K. Svore, L. Vanderwende, and C. Burges. Enhancing single-document summarization by combining ranknet and third-party sources. In Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL), 2007.
34. K. S. Thakkar, R. V. Dharaskar, and M. Chandak. Graph-based algorithms for text summarization. In Emerging Trends in Engineering and Technology (ICETET), 2010 3rd International Conference on, pages 516–519. IEEE, 2010.
35. C. Thaokar and L. Malik. Test model for summarizing hindi text using extraction method. In Information & Communication Technologies (ICT), 2013 IEEE Conference on, pages 1138–1143. IEEE, 2013.
36. S. Vijay, V. Rai, S. Gupta, A. Vijayvargia, and D. M. Sharma. Extractive text summarisation in hindi. In Asian Language Processing (IALP), 2017 International Conference on, pages 318–321. IEEE, 2017..