IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.8, Issue.4, Apr 2020 E-ISSN: 2347-2693

Deep Learning Architecture for Hybrid Multi-Document Abstractive Summarization using Sentence Embeddings

Anita Kumari Singh[1*], M Shashi[2]

[1,2]Dept. of Computer Science and Systems Engineering, Andhra University College of Engineering (A), Visakhapatnam, Andhra Pradesh -530013

[*]*Corresponding Author: anitasinghani@gmail.com.*

DOI: https://doi.org/10.26438/ijcse/v8i4.59 | Available online at: www.ijcseonline.org

Received: 03/Apr/2020, Accepted: 11/Apr/2020, Published: 30/Apr/2020

Abstract— Multi-document summarization aims at generating a comprehensive summary of multiple documents related to a common topic without repeatedly conveying the same piece of information while covering the essential information from all the documents. Extractive summarization methods exist to handle Multi-document summarization, while the Abstractive summarization methods are limited to handling single-document summaries. This paper proposes abstractive summarization of multiple documents by extending the state-of-the-art single-document abstractive summarization model Pointer-Generator to generate a multi-document summary. The short abstract summaries generated upon multiple applications of the Pointer-Generator model on individual documents are clustered at the sentence level using Skip-thought embeddings. The representative sentences from each of the clusters constitute the final summary in order to avoid similar sentences while generating the multi-document abstractive summary without loss of information. The proposed methodology is evaluated using the DUC2004 benchmark dataset and observed a gain of 2 to 7 points of ROUGE scores compared to existing state of the art methods.

Keywords—*Multi-Document Summarization, Abstractive, Skip-thought embeddings, ROUGE*

I. INTRODUCTION

The growth in information technology and its advances have catastrophically increased the volume of data generated in recent years. The huge volumes of data hence generated calls for quick and automated solutions for its effective and insightful management. With the rapidly growing data, more sophisticated solutions are also being proposed concurrently to handle the huge volumes in many other better ways while consuming lesser time.

Plenty of real-time use cases are being explored [1][2][3] on how the huge data collections could be utilized to generate unseen insights from the data and consume it to improve the performance of the organizations in a profitable and competitive way. One of the essential solutions for managing the collections of textual data is the automatic summarization of the documents.

Extractive summarization techniques select the essential parts of the original document and assemble them to generate a summary. In contrast, Abstractive summarization techniques involve synthesizing new sentences that represent the gist of the original document and are closer to a human-generated summary.

The state-of-the-art abstractive summarization approaches are limited to handling only a single document at a time. Hence, to generate a comprehensive summary of multiple articles related to a topic with opinion diversities, the authors propose two novel architectures for abstractive multi-document summarization: Multi-document Summarization using a Cascade of Abstractive and Extractive summarization [4] and the Hybrid Multi-document Abstractive Summarization using Sentence-level embeddings.

The current work generates abstractive multi-document summaries using sentence embeddings, Skip-thought vectors [5] to select the most representative sentences from the document collection without any repetition by excluding similar sentences.

II. RELATED WORK

The introductory research works on summarization were mostly based on the Extractive summarization, which has almost come to a saturation. With the inception of Deep Learning approaches using Neural Networks the entire focus of researchers in the summarization field have shifted towards the Abstractive Summarization of documents.

Some of the recent extractive summarization approaches are briefed below. Horacio et al. [6] presented an overview of the summarization models in their survey, along with the methods used for evaluating the automatic summaries. The paper [7] discusses an approach to derive extractive summarization of factual reports using deep learning in three phases, namely feature extraction, feature enhancement, and summary generation.

The paper [8] elaborates on the use of Recurrent Neural Net Language Modelling (RNNLM) for extractive broadcast News summarization. The authors of the work [9] presented an extractive single-document summarization model to extract words or sentences based on neural nets and continuous sentence features.

Some of the recent research towards abstractive summarization are: The research work [10], presents a survey on the abstractive methods for text summarization techniques divided into structured and semantic approaches. Abstractive text summarization for single-documents using Attentional Encoder-Decoder Recurrent Neural Nets is studied in work [11].

Abstractive summarization models using multi-task learning is proposed in the paper [12], where decoder parameters are shared with those of an entailment generation model, the work adopts a paradigm of sequence-to-sequence multi-task learning proposed by [13], as a baseline model.

The work [14] proposes a novel coarse-to-fine attention model. The model proposed in [15] is the beginning of successful neural machine translation that is entirely inspired on a data-driven approach. The paper discussed in [16] provides an overview of the recent developments to abstractive headline generation for documents using recurrent neural networks.

See et. al, proposed a Hybrid document summarization architecture, Pointer-Generator network [17], which can extract critical portions of the input text via pointers while generating novel sentences to build the multi-sentence gist of the input document with excellent coverage and cohesion. The work proposed in [4] makes use of the abstractive single-document summarization using Pointer-Generator and Cascades the output of multiple applications of single-document abstractive summaries using another multi-document Extractive summarization method.

III. ABSTRACTIVE MULTI-DOCUMENT SUMMARIZATION ARCHITECTURES

Abstractive summarization synthesizes novel statements to form the summary of a document and is presently limited to only single-document summarization. Single-document summarization captures the overall content from a single document to generate the summary. Multi-document summarization, on the other hand, tries to summarize a group of related documents discussing the same topic into a single summary, which includes the essential sentences from all the documents in the group.

The main challenge with multi-document summarization while generating the final summary is *coverage*, that is to captures most of the critical information in the group of documents and to avoid *redundancy* in the form of repeated information in the summary. The following sections elaborate on the two proposed Multi-document Abstractive Summarization architectures.

A. Multi-document Abstractive Summarization as a Cascade

The Multi-document Summarization as a Cascade is established to integrate the merits of Abstractive single-document summarization and Extractive multi-document summarization and achieves the benefits of both. The Multi-document Abstractive Summarization architecture works in two phases. In the first phase of the architecture, the pre-trained Pointer-Generator model [17], which is trained on the News articles from CNN/DailyMail, is used to generate the summaries for the news articles from the DUC2004 corpus.

The state-of-the-art Pointer-Generator model is used to create the shorter abstract summaries of the individual News articles from the clusters of Unifiable News articles [18]. In the second phase, multi-document abstractive summarization is applied on the shorter abstractive summaries generated in the previous phase. The architecture diagram of the Multi-document Abstractive Summarization as a Cascade of Abstractive and Extractive approach [4] is shown in Fig 1.

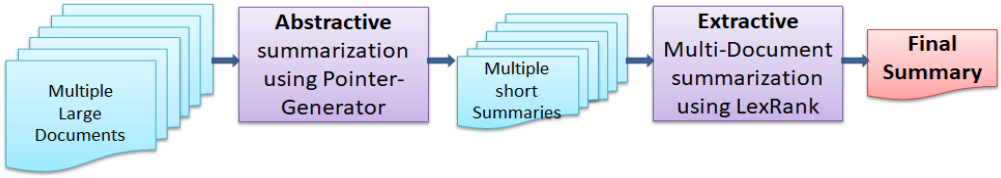

Fig 1: Multi-document Summarization Architecture.

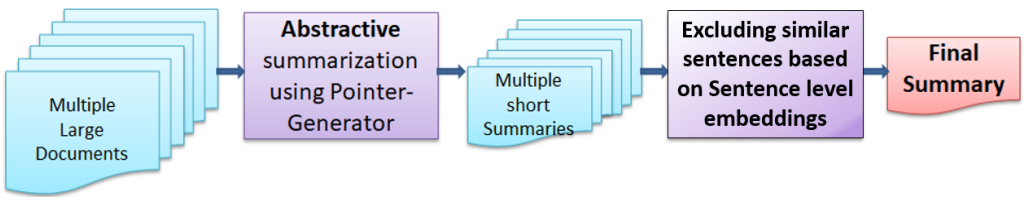

Fig 2: Hybrid Multi-document Summarization using Sentence embeddings.

B. Hybrid Multi-document Abstractive Summarization using Sentence embeddings

Hybrid Multi-document Abstractive Summarization using Sentence embeddings architecture generates a multi-sentence summary from multiple documents based on a common topic. The Hybrid Multi-document, abstractive summarization architecture, discussed below is similar to the Cascade architecture proposed earlier and works in two phases. The Hybrid Multi-document abstractive Summarization using Sentence embeddings architecture is shown in Fig 2.

1. Methodology

The first phase of the proposed architecture generates the short intermediate abstract summaries for all the documents in a cluster individually using the *Pointer-Generator* model.

Once the short intermediate abstract summaries are generated, the next phase in the architecture, groups the sentences from all the multiple summaries based on their nearness using partitional clustering on the sentence level embeddings *Skip-thought*.

Pointer-Generator: The Pointer-Generator[17] architecture extracts key portions of the input document via pointers and generates novel sentences using the generator to create the multi-sentence summary of the input document. The Pointer-Generator architecture is the state-of-the-art that generates a multi-sentence summary of text documents. The architecture overcomes the limitations of abstractive summarization methods which were limited to sentence to sentence summarization. A brief introduction to the architecture is presented in [4].

Skip-thought Vectors: Skip-thought is an unsupervised sentence encoder framework which is used to compute the sentence-level embeddings for the documents. It is an encoder-decoder based model that reconstructs the sentences based on the surrounding sentences of an encoded passage. Skip-thought works in a way that the sentences with similar semantic and syntactic representations are mapped to similar vector representations. The architecture of Skip-thought embeddings is briefed in the following Section 2.

The proposed architecture makes use of the Pointer-Generator in the first phase and Sentence-level embeddings Skip-Thought in the later phase to select the most representative sentences of the clusters of documents while avoiding similar sentences in the final multi-document abstract summary.

In the second phase the short abstract summaries hence generated using the Pointer-Generator are combined, and their sentence level embeddings using Skip-thought vectors [5] are computed. The obtained sentence vectors are clustered using the k-Medoids clustering algorithm and based on the clustering, the sentences which are most centrally located in the clusters are chosen to form the final summary of the multiple documents.

The ***k*** value in the k-Medoids clustering algorithm is the number of clusters to be formed**.** Based on the total number of sentences in all the short abstract summaries, the number of clusters to be created could be chosen. In this work, the authors choose the number of clusters based on the square root of the total number of sentences in the collection.

Once the number of clusters to be formed is decided, the sentences are clustered, and the most central representative sentences from each of the clusters are included in the final summary. The sequence of the candidate sentences in the final summary is based on the original order of the sentences in their respective clusters.

The medoids of the clusters of sentences, being the central represents the rest of the sentences of the cluster. The sentences represented by the medoids of various clusters essentially captures the semantics of all the sentences in the multiple short abstract summaries of the previous phase and hence constitute the final summary.

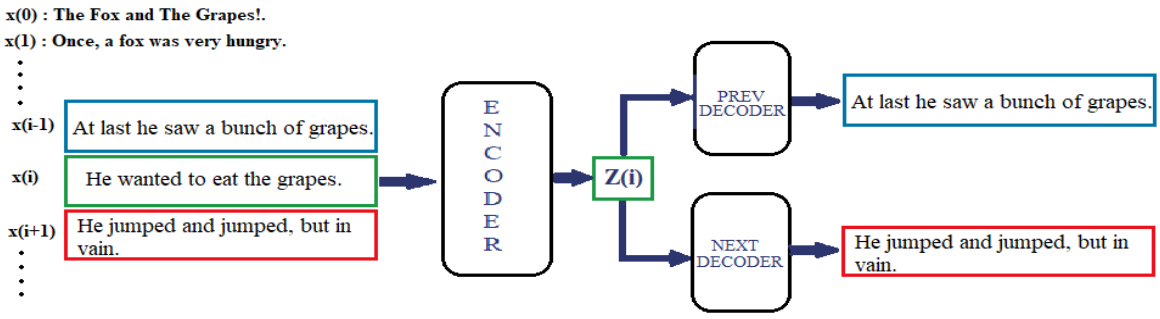

Figure 3: Skip-thought Model representation

2. Skip-thought Vectors

Skip-thought vectors makes use of the order of the sentences in the natural language to "self-supervise" and train itself. The contents in any sentence are assumed to be helpful in the better reconstruction of the neighbouring sentences. The decoders of the model are trained in a way to minimize the reconstruction error of the previous and next sentence given the embeddings, of the current sentences, **i**.

To train the Skip-thought model, the sentence tuples (s_{i-1}, s_i, s_{i+1}) consisting of the previous sentence, the current sentence, and the next sentence is used. The model representation of Skip-thought Vectors is as shown in the Fig 3.

Skip-thoughts model consists of three parts:

- **The Encoder Network**: This takes the sentence **x(i)** at index ***i*** to generate a fixed-length representation **z(i)**. It is built using a recurrent neural network like GRU (Gated Recurrent Units) or LSTM to learn the semantics of a sentence by processing the words sequentially.
- **The Previous Decoder Network**: The fixed-length representation of i^{th} sentence **z(i)** is used to generate the sentence embeddings of the $(i-1)^{th}$ sentence. The Previous Decoder Network unit is built using a recurrent neural network GRU or LSTM, to generate the sentence embedding sequentially.
- **The Next Decoder Network**: This considers the fixed-length representation of i^{th} sentence **z(i)** and tries to generate the sentence embeddings of the $(i+1)^{th}$ sentence. Again, a recurrent neural network similar to the Previous Decoder Network is used.

IV. EXPERIMENTATION AND RESULTS

A. Datasets used

CNN/DailyMail: The authors of 11], *Nallapati et. al*, generated the abstractive summaries for each News article in the stories directory of the question-answering dataset created by [19], by combining all the human-generated abstractive summary bullets for all the stories in their original order, to get a multi-sentence abstractive summary, where all bullets are treated a sentence in the summary.

DUC2004: Document Understanding Conference (DUC) 2004, consist of about 500 documents which are organized in 50 clusters, each cluster contains 10 News articles related to a common News topic form NEWSWIRE. Task 2 from DUC2004 is dedicated to generic multi-document summarization**.**

B. Generating short abstract for DUC2004

The files in the DUC2004 corpus are pre-processed as elaborated in [4], to match the file structure in CNN/DailyMail, the pre-processed DUC2004 files are tokenized using the Stanford CoreNLP tokenizer and later converted to a binary format to generate the short abstract summaries of the documents in the DUC2004 folder. The Encoder of the model captures the first 400 words in the news articles using its encoding steps and produces a summary of length 100 words in terms of decoding steps by the decoder part of the Pointer-Generator model.

The generated abstract summary files are retained in the respective folders using the same naming conventions used in the original DUC2004. Thus, in the first phase of the proposed architectures, the decoded files for each of the DUC2004 files are generated, which are the abstract short summaries for all the 500 documents in the corpus.

C. Evaluation

Recall-Oriented Understudy for Gisting Evaluation or ROUGE [20] is a Recall based metric that is used for evaluating the fixed-length summaries by making use of n-gram co-occurrence of words or phrases in summaries generated with respect to reference summaries written by humans. The effectiveness of the proposed Hybrid Multi-document Summarization using Sentence embeddings Architecture is evaluated on the DUC2004 benchmark dataset using the ROUGE metric.

ROUGE 1, ROUGE 2, ROUGE 3, and ROUGE L measures of the ROUGE metric are recorded and tabulated for evaluation. In the proposed work, the ROUGE scores of the automatically generated summaries with the existing set of four ideal human-written summaries, are calculated using the Python software package.

D. Results

The ROUGE-1, ROUGE-2, ROUGE-3, and ROUGE-L scores using the proposed architectures in comparison to LexRank [21] and MMR [22] multi-document extractive summarization models are shown in Table I. The Recall value of the ROUGE score calculated for the proposed architectures in comparison to the other methods is presented in the evaluation table.

The Multi-document Summarization as a Cascade **(MSC)** architecture has achieved 5 points improvement over LexRank while the Hybrid Summarization using Sentence embedding **(HSSE)** architecture have achieved 7 points improvement over the extractive multi-document summarization method LexRank.

Table I: Comparison table for ROUGE scores.

Algorithm	Rouge 1	Rouge 2	Rouge 3	Rouge L
HSSE	**0.45025**	**0.08962**	**0.03004**	**0.3437**
MSC	0.43013	0.08056	0.02526	0.3326
LexRank	0.38926	0.07256	0.0197	0.30853
MMR	0.32809	0.05425	0.01348	0.28925

ROUGE scores obtained by the proposed architectures its comparison with other methods is shown in Fig 4, which clearly shows the improved performance of the proposed methods over the traditional ones.

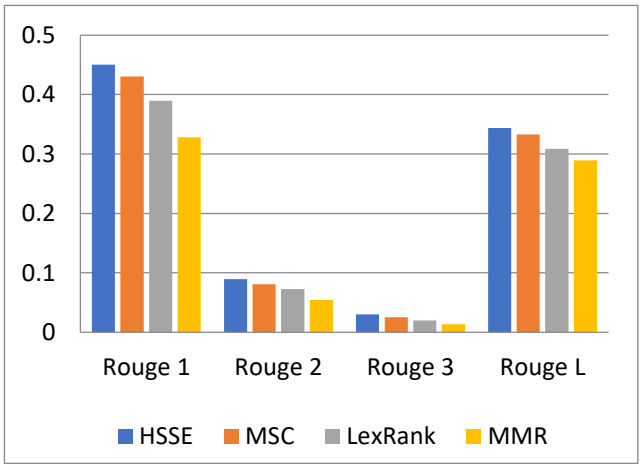

Fig 4: ROUGE Scores

The figure shows a consistent improvement of the proposed methods with respect to all the measures of ROUGE.

V. CONCLUSION

Hybrid Multi-document Abstractive Summarization using Sentence embeddings architecture generates a multi document multi sentence abstract summary in two phases. The architecture makes use of the Pointer-Generator to obtain the multiple short abstract summaries with possible redundancy in the first phase, and the Skip-thought embedding in the later phase to select the most representative sentences of the clusters of sentences upon clustering, to generate the final multi document summary.

The effectiveness of the proposed Hybrid Multi-document Abstractive Summarization using Sentence embeddings architecture is established using the ROUGE metric using the benchmark dataset DUC2004 with 7 points improvement over LexRank and 2 points improvement over the Multi-document Abstractive Summarization as a Cascade approach.

REFERENCES

[1]. Chen, CL Philip, and Chun-Yang Zhang. "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data." Information sciences 275 (2014): 314-347.

[2]. Raghupathi, Wullianallur, and Viju Raghupathi. "Big data analytics in healthcare: promise and potential." Health information science and systems 2.1 (2014): 3.

[3]. Andreu-Perez, Javier, et al. "Big data for health." IEEE journal of biomedical and health informatics 19.4 (2015): 1193-1208.

[4]. Singh, Anita Kumari, and Mogalla Shashi. "Deep Learning Architecture for Multi-Document Summarization as a cascade of Abstractive and Extractive Summarization approaches." International Journal of Computer Sciences and Engineering 7.3 (2019): 950-954.

[5]. Kiros, Ryan, et al. "Skip-thought vectors." Advances in neural information processing systems. 2015.

[6]. Saggion, Horacio, and Thierry Poibeau. "Automatic text summarization: Past, present, and future." Multi-source, multilingual information extraction, and summarization. Springer, Berlin, Heidelberg, 2013. 3-21.

[7]. Chen, Kuan-Yu, et al. "Extractive broadcast News summarization leveraging recurrent neural network language modeling techniques." IEEE Transactions on Audio, Speech, and Language Processing 23.8 (2015): 1322-1334.

[8]. Cheng, Jianpeng, and Mirella Lapata. "Neural summarization by extracting sentences and words." arXiv preprint arXiv:1603.07252 (2016).

[9]. Verma, Sukriti, and Vagisha Nidhi. "Extractive summarization using deep learning." arXiv preprint arXiv:1708.04439 (2017).

[10]. Khan, Atif, and Naomie Salim. "A review of abstractive summarization methods." Journal of Theoretical and Applied Information Technology 59.1 (2014): 64-72.

[11]. Nallapati, Ramesh, et al. "Abstractive text summarization using sequence-to-sequence rnns and beyond." arXiv preprint arXiv:1602.06023 (2016).

[12]. Pasunuru, Ramakanth, Han Guo, and Mohit Bansal. "Towards improving abstractive summarization via entailment generation." Proceedings of the Workshop on New Frontiers in Summarization. 2017.

[13]. Luong, Minh-Thang, et al. "Multi-task sequence to sequence learning." arXiv preprint arXiv:1511.06114 (2015).

[14]. Ling, Jeffrey. Coarse-to-fine attention models for document summarization. Diss. 2017.

[15]. Rush, Alexander M., Sumit Chopra, and Jason Weston. "A neural attention model for abstractive sentence summarization." arXiv preprint arXiv:1509.00685 (2015).

[16]. Shen, Shi-Qi, et al. "Recent advances on neural headline generation." Journal of computer science and technology 32.4 (2017): 768-784.

[17]. See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with Pointer-Generator networks." arXiv preprint arXiv:1704.04368 (2017).

[18]. Singh, Anita Kumari, and Mogalla Shashi. "Vectorization of Text Documents for Identifying Unifiable News Articles." corpora 10.7 (2019).

[19]. Hermann, Karl Moritz, et al. "Teaching machines to read and comprehend." Advances in neural information processing systems. 2015.

[20]. Lin, Chin-Yew. "Rouge: A package for automatic evaluation of summaries." Text summarization branches out. 2004.

[21]. Erkan, Günes, and Dragomir R. Radev. "Lexrank: Graphbased lexical centrality as salience in text summarization." Journal of artificial intelligence research 22 (2004): 457-479. [7].

[22]. Carbonell, Jaime G., and Jade Goldstein. "The use of MMR, diversity-based reranking for reordering documents and producing summaries." SIGIR. Vol. 98. 1998.

Authors Profiles

Anita Kumari Singh is a Research Scholar at the Department of Computer Science and Systems Engineering, College of Engineering (A), Andhra University, Visakhapatnam. She received her M.Tech Degree in Information Technology from Andhra University with distinction. She is having five years of research experience and worked on multiple research-based projects. She is one of the team members for two consultancy projects on Deep Learning in Natural Language Processing domain for a Japanese Software Company, Exa Wizards, TOKYO, JAPAN. She has published many technical research papers in various international journals, Noted Speaker and Thought Leader in various Technical Meetups, delivered a Tech Talk at Digital Summit 2019 organized by MiracleSoft, Lifetime member in Indian Social Science Congress (ISSC). Her areas of research interest include Data Mining, Deep Learning, Natural Language Processing, Artificial intelligence and Machine Learning.

Prof. M Shashi received her B.E. in Electrical and Electronics and M.E. in Computer Engineering with distinction from Andhra University. She received a Ph.D. in 1994 from Andhra University and received the best Ph.D. thesis award. She is a professor in the Department of Computer Science and Systems Engineering, Andhra University, Andhra Pradesh, India. Prof. Shashi was a recipient of the AICTE career award as a young teacher in 1996 and also received the Andhra Pradesh State award as the Best Teacher for Engineering stream in 2016. She is the coordinator for the Center for Data Analytics, Andhra University sponsored by ISEA Project phase II, Ministry of Electronics and Information Technology (MeitY), India. She recently completed three consultancy projects on Deep learning in NLP domain for a Japanese Software Company, Exa Wizards, TOKYO, JAPAN. Her research interests include Data Mining, Artificial intelligence, Pattern Recognition, and Machine Learning. She is a member of the Computational Intelligence group of IEEE, a life member of ISTE, CSI, and a fellow member of the Institute of Engineers (India).

