IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-3, March 2019 E-ISSN: 2347-2693

Deep Learning Architecture for Multi-Document Summarization as a cascade of Abstractive and Extractive Summarization approaches

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/v7i3.950954 | Available online at: www.ijcseonline.org

Accepted: 08/Mar/2019, Published: 31/Mar/2019

***Abstract*—**Document summarizers create a shorter compressed version of a text document automatically. Summaries are created to give the gist of the entire document covering the key points of the document with improved readability while avoiding redundancy. Abstractive summarization synthesizes summary statements of a given document and is presently limited to single document summarization. The proposed model extends the applicability of abstractive summarization for multi-document texts by proposing a new Deep Learning architecture as a cascade of Abstractive and Extractive summarization. The proposed hybrid architecture is used to generate compact and comprehensive summaries from multiple news articles published on specific topics. The architecture was evaluated using DUC 2004 data and its performance is found to be better compared to traditional Multi Document Extractive Summarization methods in terms of ROUGE scores.

***Keywords*—**Document Summarization, Abstractive, Extractive, ROUGE

I. INTRODUCTION

Data in the current era is growing at very rapid pace with billions of online documents in the form of news articles, blogs, web pages and many other are being created and shared among users every day from a variety of sources of different lengths, structures and types. This scenario leads us to solve many challenging problems associate with huge amounts of data and one among them is automatic summarization of the documents which helps to quickly and effectively find the relevance of a large document for a specific information requirement of a user.

Automatic text summarizers create a shorter and comprehensible version of large documents. Text summarization is condensing the most important information from single or multiple sources to produce a compressed version of the document useful for the particular user or users [1]. The two general approaches to automatic text summarization are Extractive text summarization and Abstractive text summarization.
Extractive summarization extracts sentence directly from the entire collection of documents depending on their importance or as per the requirements of the user without modifying the sentences themselves. Abstractive summarization involves paraphrasing some of the sentences in the source document creating new sentences which is not seen in the original document. Abstractive summarization [2] can produce better summaries compared to extractive summarization but building such models requires more complex language modeling.

Based on the type of input to the summarization models the automatic document summarization can be further classified as single document summarization or multi-document summarization. In single document summarization the most important sentences which capture the overall content of a given document constitute 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 include the essential sentences from all the documents in the group. The main challenge with multi-document summarization is redundancy and coverage aspects to avoid duplicate sentences framing the final summary.

To the best of the knowledge of the authors there is no published work on Abstractive approach to multi-document summarization which utilizes the potential of Deep learning. Till recently the success of abstractive summarization using deep learning was limited only to generating headlines [3], [4] for news articles. Deep Learning models for Multi-sentence abstractive summarizations were recently developed [5] for single documents only.

This paper proposes a hybrid approach for multi-document summarization by integrating Abstractive and Extractive approaches to achieve high quality summarization with better ROUGE scores compared to pure Extractive multi-document summarization.

The remainder of the paper is organized as follows: Section 2 elaborates the related concepts and literature used in the proposed architecture, Section 3 describes the methodology of the proposed architecture, Section 4 elaborates the experimental setup, dataset and preprocessing required. Section 5 is about the results and evaluation and the conclusions to the paper are presented in Section 6.

II. RELATED CONCEPTS AND LITERATURE

2.1 LexRank:

LexRank approach is based on sentence salience and computes a graph-based centrality scoring for sentences from multiple documents [6] related to a topic. Similarity scoring of sentences highlights the most important sentences compared to centroid based approaches that are prone to over generalization in the set of documents. Instead LexRank uses Centrality Scoring of sentences inspired by the concept of *prestige [7]* in social networks. Sentences of a document collection on a topic are related which are modeled as a weighted graph with sentences represented by individual nodes and their similarity represented by weighted undirected links. The weight of a link connecting a pair of sentences represents their cosine similarity. The sentences which are similar to many other sentences in the graph are assumed to be more central to the topic for summarization.

The PageRank [8] formula originally devised for computing web page prestige is modified as LexRank to estimate the centrality scores of sentences [6]. The sentences with centrality scores higher than a threshold are included in the summary.

2.2 Pointer-Generator:

With the advent of deep learning techniques, it is now possible to synthesize the sentences to capture the gist of the sentences making use of word embeddings already generated. The pointer- generator architecture is capable of extracting key portions of the input text via pointers while generating novel sentences to build the multi-sentence gist of the input document with good coverage and cohesion. This architecture generates multi-sentence summary of text documents as it overcomes the limitation of abstractive summarization methods being confined to only sentence to sentence summarization.

The architecture contains an encoder that captures the essence of the input document in terms of encoder states for further processing by the decoder to generate the summary word by word either by generating or selecting an appropriate portion of the input and thereby called pointer – generator model. The model building starts with generation of a global vocabulary based on the frequent words of the document corpus and their word embeddings. The decoder generates successive words of the summary by probabilistically selecting words of the vocabulary through beam search in tune with the context and decoder state.
The decoder in its successive states estimates the attention distribution over the input words, use them as weights to tailor make the encoder states for generating input features specific to the decoder state called context vector. In order to avoid repeated attention towards the most important part of the input even after inclusion in the portion of summary already generated, the decoder maintains a coverage vector to monitor the cumulative attention already given to the input words in the previous decoder states and considers the coverage vector also in addition to encoder and decoder states as input to generate attention distribution. Optimal training schedule starts with training without coverage until the weights stabilizes followed by training with coverage until the total loss which is the sum of log loss and coverage loss on validation set converges. The weights are learnt in mini-batch mode with batch size of sixteen. The words of the reference summary are fed as the input to the decoder during training phase whereas the word generated in the previous state is taken as decoder input during testing phase since no reference summary is expectable during testing.

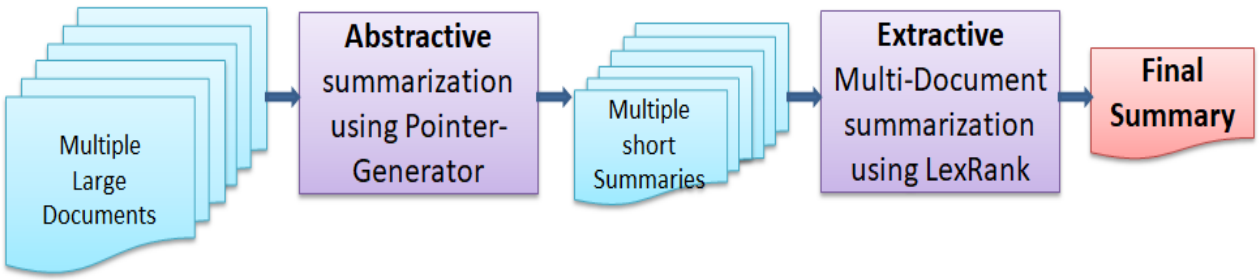

Figure 1: Framework for Hybrid summarization approach.

The encoder is implemented using a bi-directional LSTM[9] to consider the order as well as vicinity of words to extract their importance, while the decoder is implemented using (uni-directional) LSTM to add words to the summary in successive states.

III. METHODOLOGY

The proposed hybrid approach for multi-document summarization is built in two phases: The first phase makes use of the pointer-generator model originally built for single document summarization to create shorter abstract summaries of multiple news articles related to a topic. However, these abstract summaries contain considerable amount of redundancy as they were produced independently from news articles related to same topic. Hence the authors propose to apply extractive summarization that selects and assembles essential sentences from multiple summaries to form the final summary with good coverage while avoiding redundancy. The second phase of the framework generates the final summary applying LexRank, the most widely used extractive multi-document summarization method, on the abstract summaries generated during the first phase.
The framework involves cascading of abstractive summarization of multiple news articles on a topic followed by extractive summarization of multiple abstract summaries after preprocessing the text to be compatible with the input formats to generate the final summary. The effectiveness of the framework is tested on DUC 2004 corpus using ROUGE [10] scores, ROUGE 1, ROUGE 2, ROUGE 3 and ROUGE L. The performance of the framework is found to be satisfactory as the proposed hybrid approach could achieves consistently higher ROUGE scores which indicate that the proposed hybrid approach could generate better quality summaries.

The architecture diagram of the proposed hybrid summarization approach is as shown in the figure1.

IV. EXPERIMENTAL SETUP

The pre-trained model for Pointer-Generator document summarization is used in the first step of the proposed hybrid model. The Pointer-Generator model is the first step towards multi-sentence abstractive summarization. The model can be adopted for summarizing individual documents in DUC 2004 and produce multi-sentence summaries as the model itself is trained on a very large corpus of news articles using CNN and Daily Mail [11] containing more than 92k examples and 219k examples respectively. The pre-trained models are available for download.

The decoded files (intermediate summaries) produces in the first step are preprocessed as described in section 4.2 are used to produce the final summary using LexRank.

SOURCE

```
<DOC>
<DOCNO> APW19981031.0720 </DOCNO>
<DOCTYPE> NEWS </DOCTYPE>
<TXTTYPE> NEWSWIRE </TXTTYPE>
<TEXT>
At least 231 people have been confirmed dead in Honduras from former-hurricane
Mitch, bringing the storm's death toll in the region to 357, the National
Emergency Commission said Saturday. Mitch _ once, 2nd graf pvs
</TEXT>
</DOC>
```

TARGET

```
At least 231 people have been confirmed dead in Honduras from
former-hurricane Mitch, bringing the storm's death toll in the region to 357,
the National Emergency Commission said Saturday.

Mitch _ once, 2nd graf pvs
```

Figure 2: Document format for Pointer-Generator

4.1 Dataset Used:
Document understanding conference [12] (DUC) 2004 has a task dedicated to multi document summarization. Task 2 from DUC 2004 is based on generic multi document summarization consisting of 50 clusters each of which has approximately 10 news articles on the same news topic picked up from NEWSWIRE.

4.2 Pre-processing:

The DUC 2004 task 2 documents are enclosed between the <DOC> and </DOC> HTML tags. The text part from the all the documents enclosed in the <TEXT> and </TEXT> tags was extracted and converted in a form such that each sentence in the document is placed in a single line followed by a line break. All the documents are re-written in the required form by retaining their IDs with the similar naming convention as in DUC 2004. A snapshot showing the source and target text alignments is seen in figure 2. The documents

representation as shown in the target part is used by the pointer-generator model to produce the shorter abstract summaries. The model reads the first 400 words of the document using encoders and produce approximately 100 words summary using decoders phase of the model.

The shorter abstract summaries generated through pointer-generator approach are further pre-processed to remove any incomplete sentences that happen to fall into the abstract summaries.

As the length of the intermediate summaries are restricted to 100, some incomplete sentences made their way to the abstract summaries. All the incomplete sentences and very short sentences with less than 5 words in the summaries were removed and only complete and reasonably long sentences were retained and taken as input to the next phase, Extractive summarization using LexRank.

Table 1: Comparison of Rouge Scores

Algorithm	R1	R2	R3	RL
Proposed	0.43013	0.08056	0.02526	0.33260
LexRank	0.38926	0.07256	0.01970	0.30853
MMR	0.32809	0.05425	0.01348	0.28925

The figure 3 shows the ROUGE scores obtained by the proposed method and its comparison with the other method which clearly shows the improved performance of the method over the traditional methods.

V. RESULTS AND EVALUATION

ROUGE was used as the metrics for evaluating the proposed work. Recall-Oriented Understudy for Gisting Evaluation [10] or ROUGE is a Recall based metric which is used to evaluate fixed length summaries making use of n-gram co-occurrence of words/phrases in summaries generated with respect to Reference summaries generated by humans. ROUGE evaluates the generated summaries with the set of human summaries using a python software package. Different types of evaluation in ROUGE are ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S and ROUGE-SU.

The ROUGE-1, ROUGE-2, ROUGE-3 and ROUGE-L scores using the proposed method with comparison to LexRank and MMR [13] extractive summarization methods are shown in the table 1. The proposed methods have achieved 5 points improvement over one of the better performing traditional algorithms used for Extractive summarization LexRank.

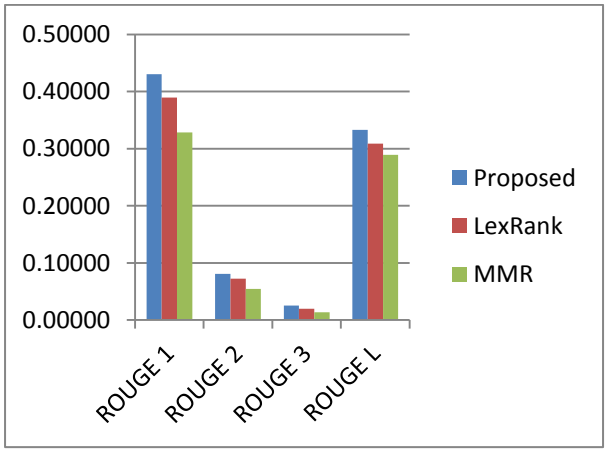

Figure 3: Rouge Scores

VI. CONCLUSIONS AND FUTURE SCOPE

The paper extends the state-of-the-art abstractive summarization architecture for multi-document summarization. The proposed architecture produces comprehensive summary on a topic by combining the Abstractive and Extractive summarization approaches in a cascade. The state-of-the-art approach for abstractive summarization using pointer-generator model is limited to single-document summarization. Summarization of multiple news articles on a topic can be handled one by one independently which results in multiple summaries for the same topic with possible redundancy. In order to avoid redundancy, the authors propose Extractive summarization of the multiple summaries as the second phase in the proposed cascade framework. The effectiveness of the framework was established using ROUGE metric.

The model uses 400 encoding steps in the training phase and hence captures the first 400 words of the documents for generating the summary, which limits the applicability of the approach for longer documents. This limitation can be overridden by splitting long documents into chunks of 400 words to capture information from the complete document.

REFERENCES

[1]. Mani, Inderjeet. Advances in automatic text summarization. MIT press, 1999.
[2]. Hahn, Udo, and Inderjeet Mani. "The challenges of automatic summarization." Computer 33.11 (2000): 29-36.
[3]. Lopyrev, Konstantin. "Generating news headlines with recurrent neural networks." arXiv preprint arXiv:1512.01712 (2015).
[4]. Nallapati, Ramesh, et al. "Abstractive text summarization using sequence-to-sequence rnns and beyond." arXiv preprint arXiv:1602.06023(2016).
[5]. See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with pointer-generator networks." arXiv preprint arXiv:1704.04368 (2017).
[6]. Erkan, Günes, and Dragomir R. Radev. "Lexrank: Graph-based lexical centrality as salience in text summarization." Journal of artificial intelligence research 22 (2004): 457-479.
[7]. Erkan, Günes, and Dragomir R. Radev. "Lexpagerank: Prestige in multi-document text summarization." Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. 2004.
[8]. Sullivan, Danny (2007-04-26). "What Is Google PageRank? A Guide for Searchers & Webmasters". Search Engine Land. Archived from the original on 2016-07-03.
[9]. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.
[10]. Lin, Chin-Yew. "Rouge: A package for automatic evaluation of summaries." Text Summarization Branches Out (2004).
[11]. Hermann, Karl Moritz, et al. "Teaching machines to read and comprehend." Advances in Neural Information Processing Systems. 2015.
[12]. Nenkova, Ani. "Automatic text summarization of newswire: Lessons learned from the document understanding conference." (2005).
[13]. Carbonell, Jaime G., and Jade Goldstein. "The Use of MMR and Diversity-Based Reranking for Reodering Documents and Producing Summaries." (1998).

Authors Profile

Anita Kumari Singh, Research Scholar, 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 and stood first in her batch. Having 5 years of research experience, worked on multiple research based projects. She is one of the team members for two consultancy projects on Deep Learning for NLP for a Japanese Software Company, Exa Wizards, TOKYO, JAPAN. Published many technical research papers in various international journals, Noted Speaker and Thought Leader in various Technical Meetups, Life time member in Indian Social Science Congress (ISSC). Her areas of research interest include Data Mining, Deep Learning, Natural Language Processing and Machine Learning.

M Shashi received her B.E. Degree in Electrical and Electronics and M.E. Degree in Computer Engineering with distinction from Andhra University. She received Ph.D in 1994 from Andhra University and got the best Ph.D thesis award. She is working as a professor in the department of Computer Science and Systems Engineering at Andhra University, Andhra Pradesh, India. She received AICTE career award as young teacher in 1996. She Received the Andhra Pradesh State award as the Best Teacher for Engineering stream in 2016. She is the coordinator for 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 for NLP for a Japanese Software Company, Exa Wizards, TOKYO, JAPAN. She published technical papers in National and International Journals. Her research interests include Data Mining, Artificial intelligence, Pattern Recognition and Machine Learning. She is a member of Computational Intelligence group of IEEE, life member of ISTE, CSI and a fellow member of Institute of Engineers (India).

