Expert Systems With Applications 211 (2023) 118442

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

DeepSumm: Exploiting topic models and sequence to sequence networks for extractive text summarization

Akanksha Joshi [a,b,c,*], Eduardo Fidalgo [a,b], Enrique Alegre [a,b], Laura Fernández-Robles [a,b,d]

[a] Department of Electrical, Systems and Automation, Universidad de León, Spain
[b] Researcher at INCIBE (Spanish National Cybersecurity Institute), León, Spain
[c] Centre for Development of Advanced Computing, Mumbai, India
[d] Department of Mechanical, Informatics and Aerospace Engineering, Universidad de León, Spain

ARTICLE INFO

Keywords:
Text summarization
Extractive
Seq2seq
Attention networks
Topic models

ABSTRACT

In this paper, we propose DeepSumm, a novel method based on topic modeling and word embeddings for the extractive summarization of single documents. Recent summarization methods based on sequence networks fail to capture the long range semantics of the document which are encapsulated in the topic vectors of the document. In DeepSumm, our aim is to utilize the latent information in the document estimated via topic vectors and sequence networks to improve the quality and accuracy of the summarized text. Each sentence is encoded through two different recurrent neural networks based on probabilistic topic distributions and word embeddings, and then a sequence to sequence network is applied to each sentence encoding. The outputs of the encoder and the decoder in the sequence to sequence networks are combined after weighting using an attention mechanism and converted into a score through a multi-layer perceptron network. We refer to the score obtained through the topic model as Sentence Topic Score (STS) and to the score generated through word embeddings as Sentence Content Score (SCS). In addition, we propose Sentence Novelty Score (SNS) and Sentence Position Score (SPS) and perform a weighted fusion of the four scores for each sentence in the document to compute a Final Sentence Score (FSS). The proposed DeepSumm framework was evaluated on the standard DUC 2002 benchmark and CNN/DailyMail datasets. Experimentally, it was demonstrated that our method captures both the global and the local semantic information of the document and essentially outperforms existing state-of-the-art approaches for extractive text summarization with ROUGE-1, ROUGE-2, and ROUGE-L scores of 53.2, 28.7 and 49.2 on DUC 2002 and 43.3, 19.0 and 38.9 on CNN/DailyMail dataset.

1. Introduction

Text summarization is one of the most significant areas of Natural Language Processing (NLP), among others such as text classification (Zhang et al., 2015), information retrieval (Al Nabki et al., 2017), named entity recognition (Al-Nabki, Fidalgo, Alegre and Fernández-Robles, 2020), translation or speech to text (Domínguez et al., 2019). In the internet era, there is a need for compact and concise representations of electronic documents that enable users to understand more information in less time. Text summarization eases this task by reducing the size of long documents and simultaneously retaining the salient information from them. Researchers have categorized text summarization as extractive or abstractive, single-document or multi-document, generic or query-focused (Cao et al., 2016; Chopra et al., 2016; Erkan & Radev, 2004; McDonald, 2007; Mihalcea & Tarau, 2004). Extractive text summarization selects the salient content or sentences from the document while leaving out the redundant and less relevant parts of the document to generate a summary (Mihalcea & Tarau, 2004). Instead, abstractive text summarization includes paraphrasing the main content of the document using natural language generation techniques (Rush et al., 2015). Single document summarization (Parveen et al., 2015) aims at summarizing a single document, whereas multi-document summarization (Erkan & Radev, 2004) uses a set of documents of a similar type to produce a summary.

Traditionally, methods for extractive text summarization are focused on human-engineered features such as word frequency features, sentence position, length, proper nouns, action nouns (Erkan & Radev, 2004; Filatova & Hatzivassiloglou, 2004). In such approaches, the sentences are scored according to these features, and the sentence selection for the summary was performed using greedy methods (Carbonell & Goldstein, 1998), graph-based approaches (Erkan & Radev,

* Corresponding author at: Centre for Development of Advanced Computing, Mumbai, India.
E-mail addresses: ajos@unileon.es (A. Joshi), efidf@unileon.es (E. Fidalgo), enrique.alegre@unileon.es (E. Alegre), l.fernandez@unileon.es (L. Fernández-Robles).

https://doi.org/10.1016/j.eswa.2022.118442
Received 5 August 2020; Received in revised form 27 July 2022; Accepted 4 August 2022
Available online 13 August 2022
0957-4174/© 2022 Elsevier Ltd. All rights reserved.

2004; Mihalcea & Tarau, 2004; Parveen et al., 2015) and optimization-based approaches (McDonald, 2007). Sentence scoring and selection approaches failed to produce a good compressed representation of the document.

Recently, approaches based on neural networks have gained momentum because of their high performance in many NLP tasks such as text classification (Al-Nabki, Fidalgo, Alegre and Aláiz-Rodrıguez, 2020), machine translation (Jean et al., 2015), text generation or question answering (Bordes et al., 2014). Several authors proposed deep learning approaches using sequence networks for extractive (Liu, 2019; Nallapati, Zhai et al., 2017; Ren et al., 2017) and abstractive (Bi et al., 2021; Li, Lam et al., 2017; Nallapati et al., 2016; Xu, Gan et al., 2020) text summarization. Despite gaining so much popularity in text summarization, neural network methods have some limitations. These methods do not capture the latent topic information in documents (Dieng et al., 2016), and thus, the summary lies in an embedding space that hardly contains any topic information from the document. Apart from this, the variants of Recurrent Neural Networks (RNN) such as Gated Recurrent Unit (GRU) (Chung et al., 2014) and Long Short Term Memory (LSTM) networks (Hochreiter & Schmidhuber, 1997) have very limited capability to retain the long-range semantics of the document (Khandelwal et al., 2018). In this work, we complemented neural networks with additional topic information from the document to take advantage of the latent content of documents, which otherwise is hardly captured using RNN. The main problem with recent state-of-the-art RNN-based summarization methods (Nallapati, Zhai et al., 2017; Zhang et al., 2018; Zhou et al., 2018) is that they fail to capture the latent topic information in the document that carries the significant content to summarize text. We aim to solve this problem by incorporating topic information in sequence networks to capture the long range semantics in the document. Also, another problem to overcome is that there are no works that eliminate redundant information in the summaries using topic distribution models per words, apart from word embeddings as representations of sentences. We make use of both sentence embeddings derived using topic and word vectors to discard redundant information and introduce diversity in the summary.

Topic modeling (Mikolov & Zweig, 2012) has been applied to capture the long-range dependencies in documents via latent topics. An increase in accuracy was reported when deep learning networks were supported with topic information (Dieng et al., 2016).

Probabilistic topic models (Blei, 2012) preserve the global semantic information in a document via latent topics that can efficiently capture the global semantic information in documents. By providing the topic information directly to RNN, the global information in the document can be preserved, avoiding the long-standing vanishing gradient problem of neural networks to remember long-term information (Pascanu et al., 2013).

To this end, to combine the merits of both approaches and increase the accuracy, we introduce *DeepSumm*, a novel summarization method which uses the global semantic information jointly with both the local syntactic and the semantic information in a document to produce summaries. LSTM networks are capable of extracting the local semantic and syntactic information as well as handling long-range dependencies to some extent. However, enriching LSTM networks with topic information enables to capture the global meaning embedded in the document, which is quite useful for generating summaries. Our proposed method obtains a summary after selecting sentences ranked using the fusion of four scores: Sentence Topic Score (STS), Sentence Content Score(SCS), Sentence Novelty Score (SNS) and Sentence Position Score(SPS).

The main contributions of this paper are:

- Firstly, we propose Deep Summarization (DeepSumm), a novel method for extractive text summarization which generates summaries through the weighted fusion of four scores – SCS, STS, SNS and SPS –. We derive STS and SCS using Sequence to Sequence (seq2seq) attention networks, whereas SNS is computed by means of the word vector representations and SPS reflects the relative positions of sentences in the documents.
- Secondly, we introduce Sentence Topic Embeddings and Sentence Content Embeddings to capture the long-range semantic dependencies and structural content information in the document. Our approach models sentences as functions of word embeddings as well as of topic distributions, and produces sentence saliency scores for both of them, SCS and STS, respectively. To derive sentence topic and sentence content embeddings, LSTM networks and Seq2Seq architectures with decoder attention are applied to generate the STS and SCS scores. Thus, we are able to calculate the saliency of sentences by using both their local and global semantic structures to retain the pertinent content in the document.
- Thirdly, a new Sentence Novelty Score (SNS) is presented to eliminate the redundant information and to introduce diversity in a summary. Our SNS uses the sentence representations derived using word and topic distribution vectors to compute a novelty score for each sentence in the document.
- Finally, The evaluation of our approach was performed on standard DUC 2002 summarization benchmark and CNN/DailyMail corpus. The DeepSumm framework achieves a very good accuracy in single-document extractive text summarization task surpassing several state-of-the-art neural network-based summarizers.

The rest of the paper is organized as follows. Section 2 reviews the related literature. Next, Section 3 presents the proposed summarization framework. In Section 4 we discuss the datasets used, experiments performed and the results obtained. Finally, Section 5 gives the conclusions drawn and the scope of future work in the field.

2. Related work

In our work we combine sentence saliency parameters to rank document sentences to generate extractive summaries which are semantically coherent. The topic and language models are exploited to identify the most significant sentences in the document.

The methods based on deep learning for text summarization recently gained momentum by achieving state-of-the-art accuracies. For extractive summarization, most of the state-of-the-art works rely on GRU or LSTM sequence networks. Regarding GRU, the following recent works are of interest. Nallapati, Zhai et al. (2017) proposed SummaRuNNER, a GRU-RNN based sequence model that can be trained extractively and abstractively to generate summaries. Their approach uses the absolute and relative position of the sentence and sentences from previously selected summaries to remove redundancy. The work of Nallapati, Zhou et al. (2017) involved two architectures – classifier and selector – consisting of GRU-RNNS for extractive text summarization, which obtained state-of-the-art performance on CNN/DailyMail and DUC 2002 datasets. Zhou et al. (2018) presented NeuSum, an end-to-end hierarchical sentence and document encoder architecture that utilize GRU-RNN to score and select sentences jointly. Their RNN-based sentence extractor takes into account previously selected sentences while estimating sentence salience to eliminate redundancy in the summary. Shi et al. (2019) introduced a novel extractive summarization framework, DeepChannel, which consists of a deep RNN-GRU for salience estimation and a salience-guided greedy sentence extraction strategy.

LSTM-based proposals include works like (Cheng & Lapata, 2016), who employed an encoder–decoder approach to extract the salient sentences and words for extractive summarization. Their encoder consists of a Convolution Neural Network (CNN) whereas their decoder architecture uses LSTM to classify sentences as summary and non-summary. Jadhav and Rajan (2018) designed SWAP-NET, to model the interactions of salient sentences and keywords in documents to produce extractive summaries. Their approach also uses a bidirectional LSTM architecture with an encoder–decoder to model the interactions between salient sentences and keywords in a document. Narayan

et al. (2018b) conceptualized extractive summarization as a sentence ranking task using LSTM-based document encoder and sentence extractor. Zhang et al. (2019) proposed HIBERT, Hierarchical Bidirectional Encoder Representations from transformer which is pre-trained using an unsupervised method to generate document encodings for extractive document summarization. Authors reported a wide improvement in performance of the system when using pre-trained HIBERT model for summarization. Wang et al. (2020) obtained local and global sentences embeddings using n-grams Convolution Neural Networks (CNN) and bidirectional LSTM networks and presented a heterogeneous graph-based neural networks for extractive summarization. Xu et al. (2020a) exploited sentence level attention in hierarchical transformer to rank sentences for unsupervised extractive summarization which in turn boosted the summarization accuracy.

Narayan et al. (2017) designed a hierarchical LSTM document encoder and an attention-based extractor with attention over side information. The side information that authors considered significant in an article is its title and image captions, along with the main body of the document. They proposed a novel training algorithm which globally optimizes the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric through a reinforcement learning objective. Zhang et al. (2018) built a latent extractive model based on a LSTM network, which instead of maximizing the likelihood of gold standard labels, directly maximizes the likelihood of human summaries given selected sentences. Tarnpradab et al. (2017) utilized hierarchical attention networks based on LSTM for extractive summarization of forum threads. Liu (2019) fine-tuned Bidirectional Encoder Representations from Transformers (BERT), a pre-trained encoder–decoder based transformer architecture to boost the accuracies for extractive summarization.

A wide variety of other approaches, including Reinforcement Learning (RL), have been applied for extractive text summarization. Feng et al. (2018) presented an Attentive Encoder Summarization (AES) which consists of an attention-based document encoder and an attention-based sentence extractor. Wu and Hu (2018) focused on introducing coherence into the neural extractive model via reinforcement learning. Their neural coherence model is composed of a word-level CNN encoder and a bidirectional GRU sentence level encoder, allowing to capture the cross-sentence local entity transitions. Yao et al. (2018) applied deep reinforcement learning for extractive text summarization in which they used Deep Q-Network (DQN). Their sentence encoder consisted of CNN and document encoder modeled using bidirectional GRU. It can model salience and redundancy of sentences in the Q-value approximation and learn a policy that maximizes the ROUGE score with respect to gold summaries. Zhong et al. (2019) explored combination of several networks for extractive summarization and found that BERT-based LSTM-pointer networks further optimized by RL gives the best accuracy.

Zhong et al. (2020) formulated the extractive summarization task as a semantic text matching problem and propose a novel summary-level framework utilizing BERT to match the source document and candidate summaries in the semantic space. Bi et al. (2021) introduced AREDSUM-SEQ redundancy-based conditional sentence order generator network to score and select sentences by jointly considering their salience and diversity within the selected summary sentences. Their second model, AREDSUM-CTX, used an additional sentence selection model to learn to balance the salience and redundancy of constructed summaries. To capture long-range dependencies throughout a documents, Xu, Gan et al. (2020) presented a DISCOBERT, a discourse-aware neural summarization model to extract sub-sentential discourse units as candidates for extractive selection on a finer granularity and structural discourse graphs are constructed based on RST trees. Some unsupervised approaches have also been introduced by Joshi et al. (2019), Li, Lam et al. (2017), Li, Wang et al. (2017), Liu et al. (2021) and Xu et al. (2020b) for extractive summarization.

Though RNN-based models can remember long context, such large-scale networks are unable to capture the global information present in long documents (Pascanu et al., 2013; Sutskever et al., 2013) which can otherwise be encapsulated through topic models. Topic models have been used earlier for improving the sequence networks (Dieng et al., 2016; Lau et al., 2017; Le & Mikolov, 2014; Mikolov & Zweig, 2012; Wang, Orton et al., 2018) but these models fail to capture the structural content of text. Ghosh et al. (2016) presented contextual LSTM models, which incorporate the topic information of the text in LSTM networks. Ji et al. (2016) explored multi-level recurrent architectures by efficiently leveraging document context information in language models. Tang et al. (2019) tried to combine a sequence modeling component with topic modeling component to contain the semantics and sequential structure of the texts for text generation (Tang et al., 2019).

However, none of the approaches based on deep neural network for text summarization made use of topic information to encode the documents and sentences in them. The approaches indicated above utilized word embeddings to represent the documents but missed on the global information content which can be captured using topic vectors. In our encoder–decoder framework, we fuse the information obtained from both word embeddings and topic vectors. Our encoder–decoder framework is also different from previous works, as we encode our sentences using LSTM networks and produce sentence content and sentence topic embeddings. Then, sequence to sequence LSTM network is applied to generate score for sentences. We use the scores from the LSTM network and finally fused them with other sentence scores to classify sentences as summary/non-summary. Moreover, to the best of our knowledge, there are no works that eliminate redundant information in the summaries using topic distribution models per words, apart from word embeddings as representations of sentences. In our work, we utilized both sentence content and sentence topic embeddings in the Sentence Novelty Parameter to produce non-redundant and diverse summaries.

Wu et al. (2017) presented an importance evaluation function using topics to generate single document summaries. Issam et al. (2021) utilized topic clusters in the document to generate sub-documents based on topics and then applied TextRank to produce final summaries from sub-documents. Authors (Zheng et al., 2020) presented a method to embed topic model component into seq2seq model, by using the last hidden state from the encoder to infer topic information and incorporate the topic-level features for abstractive summarization. Our framework differs from them as we embed the topic information obtained using Latent Dirichlet Allocation (LDA) in the sequence networks rather obtaining any topic information directly from sequence models. Wang, Yao et al. (2018) used topic level and word level attention in Convolutional Sequence to Sequence networks for abstractive summarization. Their approach is distinct from ours as they only incorporated topic information as attention weights whereas we generated sentence topic embeddings to attend to topic information in document apart from word level information. Xiao and Carenini (2019) focused on extracting local and global content from the document leveraging topic information but the extracted topic from LSTM minus network.

Narayan et al. (2018b) proposed topic-conditioned convolution Seq2Seq networks for extreme summarization – one line summaries – of news articles. They experimentally demonstrated that convolution layers capture long-range dependencies in document better than RNNs, which is useful to perform document level abstraction and inference. Though they utilized topic information in their framework, their approach is different from ours because they used topic information with CNN networks to generate one line abstractive summaries of documents. However, our framework is focused on extractive summarization using sequence networks rather than convolution networks. Mehta et al. (2018) proposed LSTM based sequence encoders that jointly use topic models to learn attention weights across sentence words to produce abstracts of scientific articles. Their approach is quite different

Table 1
Framework parameters.

S.no	Acronym	Parameters
1	T_j	Topic vector of jth word
2	x_j	Word embedding of jth *word*
3	E_{wi}	Sentence Content Embeddings of ith sentence
4	E_{Ti}	Sentence Topic Embeddings of ith sentence
5	SCS	Sentence Content Score
6	SPS	Sentence Position Score
7	SNS	Sentence Novelty Score
8	FSS	Final Sentence Score
9	Seq2seq	Sequence to Sequence
10	MLP	Multi-Layer Perceptron

from ours as they used modified Latent Dirichlet Allocation (LDA) to generate document context embeddings, whereas we make use of both LDA and sequence networks to generate sentence and document vectors based on topic and word information. This gives an edge over the document context encapsulated using LDA model only.

3. Proposed DeepSumm method

3.1. Problem formulation

We formulate the problem of extractive text summarization as a combination of a sentence scoring and a selection problem. Each sentence in the document is ranked based on its relevance according to the assigned score, and then, a given number of the top-ranked sentences are selected to form the summary. Given a document D made up by a sequence of N sentences $S_1, S_2, \ldots, S_N$ and sequence of M words as $w_1, w_2, \ldots, w_M$, the summary is generated by a subset of N that contains the most relevant sentences in the document. The relevance of the sentences are determined based on their structural content, topic information, relative position and novelty of that sentence in the document.

In this work, the self-attention sequence networks (Vaswani et al., 2017) are employed to encode the information in the document. They are appropriate for identifying local structural context because of their sequential nature. The overall pipeline of the proposed DeepSumm framework is illustrated in Fig. 1. The parameters used in the DeepSumm framework are given in Table 1. In the subsequent sections, we describe the key components of the proposed deep learning framework for the generation of an extractive single-document summary.

3.1.1. Probabilistic topic distribution per word

Probabilistic topic models were proposed by Blei (2012) to capture the global semantic structure of the documents. The main objective of topic modeling methods is to model documents as collections of multiple latent topics. Each topic can be seen as a distribution of semantically coherent terms and each document exhibits these topics with different probabilities or proportions. One of the probabilistic topic models is LDA (Blei et al., 2003), whose main goal is to find the K latent topics $T = \{T_1, T_2, \ldots, T_k\}$ in a collection of documents where each topic is a collection of words that tend to co-occur together. LDA is better than other topic models—Latent Semantic Analysis (LSA) (Landauer et al., 1998) and Probabilistic Latent Semantic Analysis (pLSA) (Hofmann, 1999) as LDA generalizes well for new documents and has less risk of over-fitting. We use LDA to generate topic vectors T_D for each document present in the distribution, and topic vectors for each word, as t_{w_j} for the jth word, w_j, in a document. In this work, we consider topic vectors for each word T_j as point-wise addition of the word topic vector t_{w_j} generated by LDA, plus the document topic vector t_D, as: $T_j = t_{w_j} + t_D$.

3.1.2. Word embeddings

The word embeddings for each word are computed in the document to capture the structural content information. The pre-trained word vectors (Pennington et al., 2014) are used to represent each word as x_j in d-dimensional embedding space, $R^{M\times d}$.

3.1.3. Sentence encoder

We encode topic vectors per word, T_j and word vectors, x_j computed using pre-trained embeddings, as sentence vectors by means of two bi-directional LSTMs. On the one hand, a bidirectional LSTM takes the topic vectors of each word, T_j, in a sentence as input to extract the sentence embedding, termed as E_{Ti}, which relates to the topic information of ith sentence. The forward LSTM reads the sentence S_i from T_{i1} to T_{im} and the backward LSTM from T_{im} to T_{i1}. E_{Ti} is produced by concatenating the final hidden output states, $\overrightarrow{h_{Tt}}$ and $\overleftarrow{h_{Tt}}$ of the forward and backward LSTMs as stated in Eqs. (1), (2) and (3).

$$\overrightarrow{h_{Tt}} = \text{LSTM}(T_t, \overrightarrow{h_{T(t-1)}}) \tag{1}$$

$$\overleftarrow{h_{Tt}} = \text{LSTM}(T_t, \overleftarrow{h_{T(t+1)}}) \tag{2}$$

$$E_{Ti} = [\overrightarrow{h_{Tt}}, \overleftarrow{h_{Tt}}] \tag{3}$$

On the other hand but similarly, word embeddings, x_{im}, of a sentence i, are inputted to another bidirectional LSTM to extract the sentence embedding, E_{wi}. The forward LSTM for producing E_{wi} reads the sentence S_i from x_{i1} to x_{im} and the backward LSTM from x_{im} to x_{i1}. Eqs. (4)–(6) indicate the calculation of sentence embeddings E_{wi}.

$$\overrightarrow{h_{xt}} = \text{LSTM}(x_t, \overrightarrow{h_{x(t-1)}}) \tag{4}$$

$$\overleftarrow{h_{xt}} = \text{LSTM}(x_t, \overleftarrow{h_{x(t+1)}}) \tag{5}$$

$$E_{wi} = [\overrightarrow{h_{xt}}, \overleftarrow{h_{xt}}] \tag{6}$$

3.1.4. Sentence content and topic saliency extractor

As in the previous Section 3.1.3, two similar pipelines are designed for computing sentence salience and scores from sentence vectors; one for sentence topic embeddings, E_{Ti}, and the other one for sentence embeddings, E_{wi} based on word vectors. The Sequence to Sequence (seq2seq) attention networks are employed to obtain the sentence saliency based on topic and word vectors. The proposed Seq2Seq architecture consists of a LSTM encoder that reads the sentences one by one and a LSTM decoder that tries to generate the target sequence through an attention mechanism (Bahdanau et al., 2015). The objective of the encoder is to derive a document representation based on the sentences and words present in it.

In the following, we formulate the pipeline that inputs sentence embeddings, E_{wi} obtained using word vectors in the previous section. The encoder consists of a bidirectional LSTM that takes sentence embeddings E_{wi} as input to generate an encoded document representation as described in Eqs. (7) and (8).

$$\overrightarrow{h_{E_w i}} = \text{LSTM}_{\text{encoder}}(E_{wi}, \overrightarrow{h_{E_w(i-1)}}) \tag{7}$$

$$\overleftarrow{h_{E_w i}} = \text{LSTM}_{\text{encoder}}(E_{wi}, \overleftarrow{h_{E_w(i+1)}}) \tag{8}$$

The decoder is also composed by a bi-directional LSTM that takes the sentence embeddings and attention weighted encoder outputs into consideration to produce decoder hidden states, $\overrightarrow{h_{D_w i}}$ and $\overleftarrow{h_{D_w i}}$ as given in Eqs. (9) and (10).

$$\overrightarrow{h_{D_w i}} = \text{LSTM}_{\text{decoder}}(E_{wi}, \overrightarrow{h_{D_w(i-1)}}) \tag{9}$$

$$\overleftarrow{h_{D_w i}} = \text{LSTM}_{\text{decoder}}(E_{wi}, \overleftarrow{h_{D_w(i+1)}}) \tag{10}$$

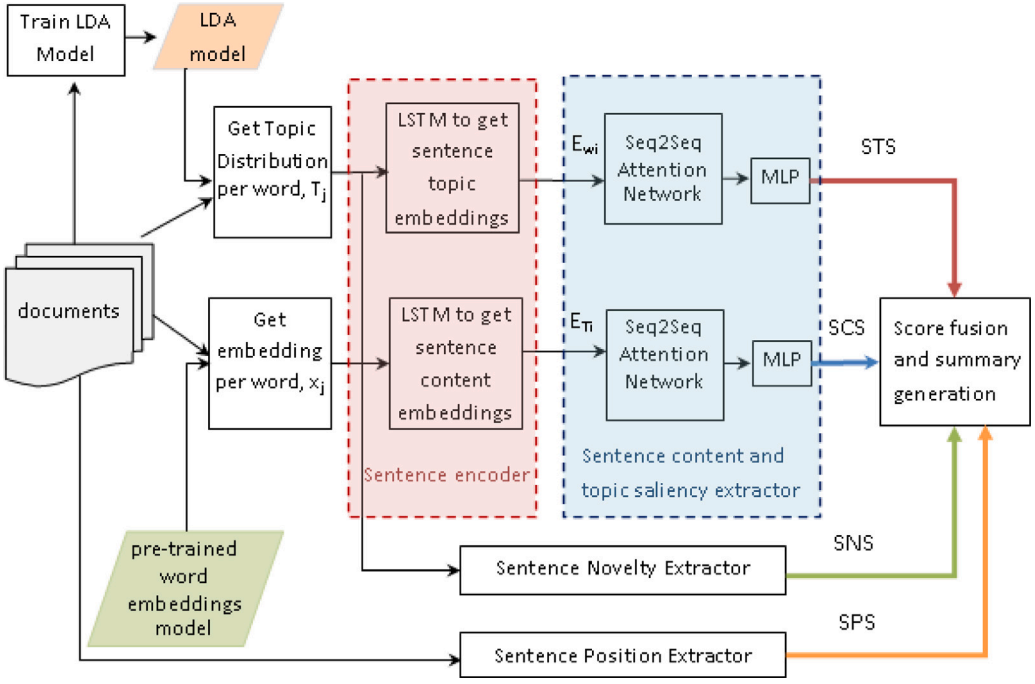


Fig. 1. Schema of the DeepSumm architecture.

The encoder and decoder outputs e_i, d_i of our pipeline consist of the following hidden states as given in Eqs. (11) and (12), respectively.

$$e_i = (\overrightarrow{h_{E_{wi}}}, \overleftarrow{h_{E_{wi}}}) \tag{11}$$

$$d_i = (\overrightarrow{h_{D_{wi}}}, \overleftarrow{h_{D_{wi}}}) \tag{12}$$

Then, an attention mechanism is applied to find the global sentence saliency for each sentence using the following Eqs. (13) and (14).

$$\alpha_{ij} = \frac{\exp(d_i, e_j)}{\sum_{j=1}^{N} \exp(d_i \cdot e_j)} \tag{13}$$

$$\overline{e_i} = \sum_{j=1}^{N} \alpha_{ij} \cdot e_j \tag{14}$$

where α_{ij} is a scalar value indicating the importance of ith sentence and $\overline{e_i}$ is the weighted sum of sentence vectors.

The decoder and attention weighted encoder outputs are finally fed into a MLP network to generate scores for each sentence in the document as in Eqs. (15) and (16).

$$a_i = \text{ReLU}(U \cdot [\overline{e_j}; d_i] + u) \tag{15}$$

$$P(y_i = 1 | E_{wi}) = \sigma(V \cdot a_i + v) \tag{16}$$

where, U, V are the learned weights of the encoder and decoder, respectively, and u, v are the bias parameters of the encoder and decoder, respectively. ReLU is a Rectified Linear Activation function that output the input directly if the input is positive otherwise, 0.

Thus, Sentence Content Score (SCS) is computed using Eq. (17)– by inputting the sentence embeddings E_{wi} to the pipeline described in this section. The Sentence Topic Score (STS) is obtained as given in – Eq. (18)– by inputting the topic distribution sentence encodings E_{Ti} to another seq2seq network designed for encoding sentence topic vectors. STS is able to capture the global semantics in the document whereas, using SCS, we are able to apprehend the local syntactic information in the document.

$$\text{SCS}_i = P(y_i = 1 | E_{wi}) \tag{17}$$

$$\text{STS}_i = P(y_i = 1 | E_{Ti}) \tag{18}$$

3.1.5. Sentence novelty extractor

We propose a new Sentence Novelty Score (SNS) that progressively scans the document one sentence at a time and assigns a score to each sentence depending on the novelty of the sentence with respect to all the previous ones. The novelty of each sentence is calculated based on the sentence embeddings E_{wi} and the topic distribution sentence encodings E_{Ti} as given in Eq. (19).

$$\text{SNS}_i = \begin{cases} 1 & \text{if } i = 1 \\ \frac{1}{i-1} \sum_{j=1}^{i-1} \frac{(1-(\text{Sim}(E_{wi}, E_{wj}) + \text{Sim}(E_{Ti}, E_{Tj})))}{2} & \text{otherwise} \end{cases} \tag{19}$$

where, $\text{Sim}(x, y)$ is the cosine similarity between vectors x and y. SNS_1 is set to 1 considering the first sentence of the article as the most significant and novel to be included in the summary. To obtain the sentence novelty, both sentence content and topic representations as generated in Section 3.1.3 are used. Through sentence content embeddings, we can find the sentences which are semantically similar to each other and thus can eliminate redundancy in summary. Enriching the novelty calculation with sentence topic embeddings, those sentences in summary can be discarded which discuss about similar topics and sometimes, may not be captured with sentence content embeddings only. Experimentally on small test dataset, it was found that average of both scores work better in computing novelty score for each sentence. The SNS is low for redundant sentences in the document, and it is robust enough to introduce diversity in the summary by producing a high score for sentences which are not covered in the previous text of the document.

3.1.6. Sentence position extractor

In news documents, the sentences which occur earlier in the document are deemed more significant in comparison to other sentences in the document (Edmundson, 1969; Luhn, 1958). Therefore, our Sentence Position Score (SPS) assigns to each sentence a relative score based on its relative position on the document and computed as given in Eq. (20).

$$\text{SPS}_i = \frac{N - P_i}{N} \tag{20}$$

where, P_i is the absolute position of sentence, i in the document. The SPS will assign higher scores to the sentences which are in the

Table 2
Databases information. # stands for 'number of'.

Dataset	Type	Usage	# Documents	# Categories
CNN/DailyMail	News	Training	287,227	–
		Validation	13,368	–
		Testing	11,490	–
DUC 2002	News	Testing	567	59

beginning of the document compared to those which occur in later part of the document.

3.1.7. Scores fusion and summary generation

We finally fused SCS, STS, SNS and SPS to obtain a final sentence score (FSS) for a sentence i, as given in Eq. (21).

$$\mathrm{FSS}_i = \alpha \cdot \mathrm{SCS}_i + \beta \cdot \mathrm{STS}_i + \gamma \cdot \mathrm{SNS}_i + \delta \cdot \mathrm{SPS}_i \tag{21}$$

In FSS_i, the sentence with highest score is considered as the most significant to be included in the summary. The values of α, β, γ, and δ are determined empirically. Finally, the sentences of a document are arranged in descending order with respect to their FSS and the top k sentences or words of the list are picked to form the extractive text summary of the document.

It should be noted that the time complexity for the computation of SCS, STS, and SPS scores is O(N), assuming that we are given a document with N sentences to be summarized. In the case of SNS calculation, we compute the cosine similarity of each sentence with all the sentences preceding it in the document. The cosine similarity is calculated based on the sentence embeddings, E_{wi}, and the topic distribution sentence encodings E_{Ti} as given in Eq. (19). Hence, the time complexity of sentence novelty calculation using sentence embeddings becomes O(N(N-1)/2) and, similarly, the complexity using topic distribution sentence encodings is O(N(N-1)/2). Since both of them can be computed in the same loop/pass, the overall complexity of SNS calculation can be given as O(N(N-1)/2).

4. Experimental analysis and results

4.1. Datasets

For the supervised – training – of our method, we need a large annotated dataset for text summarization. CNN/DailyMail (Hermann et al., 2015) is the biggest dataset that contains news articles and is frequently used in question-answering research. CNN and DailyMail comprise 197,000 and 90,000 stories, respectively. As extractive summaries of news documents are not available, we utilize the highlights, which are actually abstractive summaries, given along with the news articles to produce their extractive summaries. Those sentences are greedily added from the document to the gold summary that maximizes the ROUGE-1 and ROUGE-2 scores when matching them with the highlights. A similar approach was followed by Cheng and Lapata (2016) to obtain summaries from CNN/DailyMail datasets to train their extractive summarization methods. The standard train, test and validation split for the dataset as given in Table 2 are used in this work. For validating our approach on a different dataset, we also make use of the standard summarization benchmark dataset DUC2002. DUC 2002[1] was created by the National Institute of Standards and Technology (NIST) for Document Understanding Conferences (DUC) to evaluate and analyze the advances in the field of text summarization. The DUC 2002 dataset consists of 567 news articles from 59 news categories. There are two or more human summaries given for each of the news articles.

4.2. Experimental set up

We first split the document into sentences and tokenized them into words. For this purpose, we used the sentence and word tokenizer functionality from the freely available Natural Language Processing Toolkit (NLTK).[2] The words were represented by means of 100-dimensional GloVe embeddings (Pennington et al., 2014). The length of topic vectors for each word and document extracted using LDA is 432. For LDA, different dimensions were tried on CNN/DailyMail validation set and best accuracy is reported at 432 dimensions. The size of the hidden layer of LSTM was set to 256 and of MLP to 128. We used a 0.0001 learning rate and employed gradient clipping of ± 0.5. The learning rate was initially set to 0.01 and reduced by a factor of 10 first after 50th iteration and then after 75th iteration. The batch size was kept 64 and we trained our network using stochastic gradient descent and Adam optimization algorithm (Kingma & Ba, 2014). The dropout probability of 0.5 have been applied in the encoder and 0.25 in the decoder. Our network was trained for a maximum of 100 epochs and the best model was selected based on validation accuracy metric. We set values of α, β, γ and δ in Eq. (21) to 0.45, 0.45, 0.05 and 0.05 respectively. The values of these parameters were determined empirically on a set of 5000 news documents randomly selected from the CNN/DailyMail validation data. To determine the optimal parameters, a grid search is performed over the values of α, β, γ and $\delta \in [0, .05, \ldots, 0.95, 1]$ with $\alpha + \beta + \gamma + \delta = 1$, which outputs the feasible combinations. The parameter values which yielded the highest values of average over Recall-Oriented Understudy for Gisting Evaluation scores (ROUGE) (Lin, 2004) were selected as final values for α, β, γ and δ as given in Eq. (22)

$$\{\alpha, \beta, \gamma, \delta\} = \mathrm{argmax}_{(\alpha,\beta,\gamma,\delta \epsilon [0,0.05,\ldots,1])} \frac{\text{ROUGE-1 } + \text{ROUGE-2}}{2} \tag{22}$$

where, ROUGE-1 and ROUGE-2 are the evaluation metrics used for evaluating summaries (Lin, 2004). The experiments have been carried out in a machine with 2 T K40M GPUs with 12 GB memory each, Intel Xeon processor with 3.00 GHz frequency, and 64 GB RAM.

4.3. Evaluation

The ROUGE-1, ROUGE-2 and ROUGE-L metrics (Lin, 2004) have been used to evaluate and compare our approach with other state-of-the-art methods. ROUGE metrics are computed by matching unigrams, bigrams and the longest common subsequences between the system and gold summaries. For DUC 2002 dataset, we kept the summary length to 100 words, and for CNN/DailyMail dataset full-length ROUGE metric is used to ease the comparison of our results with other approaches.

We selected the following state-of-the-art methods to carry out a comparative analysis of the results achieved with our method. For both datasets, we considered the following methods:

NN-SE (Cheng & Lapata, 2016) consists of a hierarchical document encoder and attention-based content extractor that jointly scores and select sentences to generate extractive summaries.

SummaRuNNer (Nallapati, Zhai et al., 2017) is a simple RNN-based sequence classifier for extractive summarization. It employed a novel training mechanism to train the network using abstractive summaries.

HSSAS (Al-Sabahi et al., 2018), is a general neural network-based approach that extracts sentences from a document by treating summarization problem as a classification task. Their network follows a hierarchical structure to reflect the hierarchical nature of documents and used two levels of self-attention mechanism to attend more important content for summarization.

LEAD baseline selects the first three leading sentences from the document to produce a summary for comparison.

[1] https://duc.nist.gov/data.html.

[2] https://www.nltk.org/.

DeepSumm-content is our deep summarization framework where only sentence content embeddings have been used for evaluating the summarization framework.

DeepSumm-topic is our deep summarization framework where we utilized only topic information and embeddings in the document to perform a comparative evaluation of our system with DeepSumm where both sentence content and topic embeddings have been exploited for summary generation.

Specifically for CNN/DailyMail dataset, the results are also reported on following methods:

REFRESH (Narayan et al., 2018a), that globally optimize the ROUGE evaluation metric rather cross entropy objective and produces extractive summaries using learning to rank sentences via a reinforcement learning algorithm.

Bi-AES (Feng et al., 2018) is an attentive bi-directional encoder based extractive summarization technique to generate summaries. Bi-AES can generate a rich document representation by considering both the global information of a document and the relationships of sentences in the document

RNES (Wu & Hu, 2018) is a neural coherence model to capture the cross-sentence semantic and syntactic coherence patterns. The model obviates the need for feature engineering and can be trained in an end-to-end fashion using unlabeled data. The RNES model learns to optimize coherence and informative importance of the summary simultaneously using reinforcement learning.

NeuSum (Zhou et al., 2020) is a neural network framework for extractive summarization that jointly scores and select the summary sentences. It uses document encoder to encode each sentence of the document and then RNN-based sentence extractor to score sentences with their representations while remembering the partial output summary.

BERTSum (Liu, 2019) is an extractive summarization technique that fine-tuned BERT architecture for extractive summarization. Authors tried several summarization layers over BERT architecture and found BERTSum with inter-sentence transformer layers achieve the best performance.

PACSUM(BERT) (Zheng & Lapata, 2019) is an unsupervised summarization algorithm that employed BERT to capture sentence similarity and built graphs with directed edges arguing the contribution of any two nodes to their respective centrality is influenced by their relative position in document.

DASG (Liu et al., 2021) presented a graph-based single-document unsupervised extractive method that creates a distance-augmented sentence graph (DASG) from a document that enables fine-grained modeling of sentences and characterize the original document structures.

JECS (Xu & Durrett, 2019) consists of a sentence extraction model joined with a compression classifier that decides whether or not to delete syntax-derived compression for each sentence.

HIBERT (Zhang et al., 2019) is an unsupervised approach that used Hierarchical Bidirectional Encoder Representations from Transformers for document modeling and then pre-trained HIBERT model is exploited for achieving document summarization.

PNBERT+RL (Zhong et al., 2019) is a supervised extractive summarization technique based on the combination of several networks such as LSTM Pointer networks, pre-trained with BERT and further optimized by Reinforcement Learning to improve the accuracy.

STAS + PACSUM (Xu et al., 2020a) is a Sentence Level Transformer based Attentive Summarization that combines the ranking of sentences with the ones obtained from PACSUM network to improve the summarization performance.

AREDSUM (Bi et al., 2021) are redundancy-aware iterative ranking methods for extractive summarization extending BERTSUMEXT (Liu, 2019).

HSG (Wang et al., 2020) is a heterogeneous graph-based neural network for extractive text summarization.

DISCOBERT is a discourse-aware extractive summarization system that leverages two types of discourse graphs as inductive bias to capture long-range dependencies among discourse units.

MATCHSUM (Zhong et al., 2020) is a novel summary-level framework to match the source document and candidate summaries in the semantic space.

Whereas for DUC 2002 dataset, we took into consideration:

Integer linear programming (ILP) is a phrase-based summarization system proposed by Woodsend and Lapata (2012) that attempts to cover multiple aspects of summarization such as content selection, surface realization, paraphrasing, and stylish conventions. These features are learned separately using specific "expert" predictors but are optimized jointly using ILP model to generate summaries.

Egraph (Parveen & Strube, 2015) is an entity graph-based method for extractive single-document summarization that considers importance, non-redundancy and local coherence simultaneously. The input documents are represented by bipartite graph, and sentences are ranked based on importance by applying a graph-based ranking algorithm.

Tgraph (Parveen et al., 2015) is another unsupervised entity graph-based system, wherein the nodes are represented using topics rather entities, and the graph is weighted and dense as compared to Egraph method (Parveen & Strube, 2015).

URANK (Wan, 2010) is a unified rank methodology that simultaneously performs single and multi-document summarization. The mutual influences between the two tasks are incorporated into a graph model and the ranking scores of a sentence for the two tasks can be obtained in a unified ranking process.

CoRank (Fang et al., 2017) is an unsupervised summary extraction method that combines word–sentence relationship into the graph-based ranking model, such that the mutual influence is able to convey the intrinsic status of words and sentences accurately.

SummCoder (Joshi et al., 2019) is an auto-encoder based unsupervised extractive summarization method. Authors did a weighted fusion of sentence scores based on its saliency derived using auto-encoders, sentence position and novelty parameter to get the final scores for ranking sentences for generating extractive summaries.

4.4. Results

As shown in Table 3, DeepSumm achieved very good accuracy for the task of single-document extractive summarization on DUC 2002 dataset. The ROUGE-1, ROUGE-2 and ROUGE-L scores of 53.2, 28.7 and 49.2 yielded by DeepSumm outperformed all the considered state-of-the-art approaches. None of the state-of-the-art RNN-based summarization approaches such as NN-SE, SummaRuNNer, SummCoder, HSSAS utilizes latent topic information in the document which makes our proposed method superior to them. This supports the efficacy of our proposed framework that utilizes both topic distribution vectors and language models to derive extractive summaries of the document. A comparative evaluation on our DeepSumm-topic and DeepSumm-content also shows that both topic and word embeddings carry complementary information. The combination of the two increases the accuracy of the summarization system.

A comparative analysis of the performance of different sentence scores, SCS, STS and FSS, from 20 randomly selected documents on DUC 2002 dataset is illustrated in Fig. 2. ROUGE-1, ROUGE-2 and ROUGE-L metrics were computed when raking and extracting the sentences of the documents to generate the extractive summaries using SCS and STS, besides the default score FSS. It can be seen that STS, which is computed using topic distribution sentence encodings, achieved as good ROUGE scores on the documents as SCS, which is based on word embeddings. Even in some documents, STS yielded higher ROUGE scores than SCS. It depicts that probabilistic topic distribution encodings are capable of extracting the latent topic information of the document, which is complementary to the information captured using word embeddings. The global semantic information encapsulated using topic distribution encodings is quite relevant for generating good summaries and can contribute towards better summarization systems. The final sentence score – FSS – generated using the fusion of SCS, STS, SNS and SPS scores

Table 3
Comparative analysis of DeepSumm with state-of-the-art algorithms on DUC 2002.

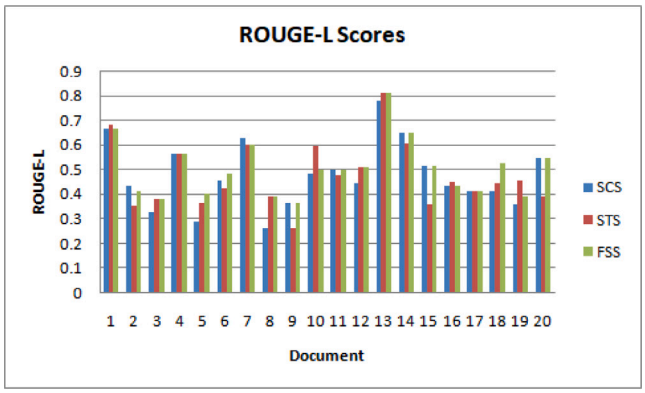
Method	ROUGE-1	ROUGE-2	ROUGE-L
LEAD	43.6	21.0	40.2
ILP	45.4	21.3	42.8
NN-SE	47.4	23.0	–
SummaRuNNer	47.4	24.0	14.7
Egraph + coh	47.9	23.8	–
Tgraph + coh	48.1	24.3	
URANK	48.5	21.5	–
SummCoder	51.7	27.5	44.6
HSSAS	52.1	24.5	48.8
CoRank	52.6	25.8	
DeepSumm-content	52.1	27.7	48.3
DeepSumm-topic	52.7	28.5	48.8
DeepSumm	**53.2**	**28.7**	**49.2**

Table 4
Gold summary and DeepSumm generated summary for a document from DUC 2002 dataset.

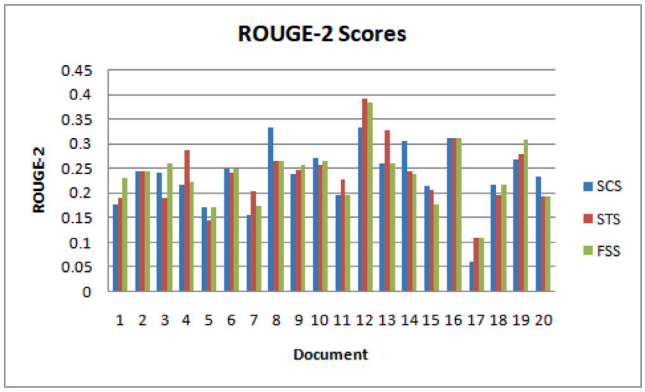
Gold summary
President Bush named career diplomat Deane Hinton as ambassador to Panama as a recess appointment since Congress is not in session. Hinton, currently ambassador to Costa Rica, replaces Arthur Davis who had been recalled in protest of what the administration\considered the stealing of the Panamanian elections by General Manuel Noriega. Davis was later returned to Panama after US forces invaded Panama and Guillermo Endara was installed as president. Hinton has also been ambassador to El Salvador and Pakistan. Senate Majority Leader George Mitchell called Hinton highly qualified because of his "wide-ranging experience and expertise in Central America".
DeepSumm summary
President Bush has named career diplomat Deane Hinton as ambassador to Panama, the White House announced Tuesday. Hinton, currently ambassador to Costa Rica, replaces Ambassador Arthur H. Davis, who was recalled by Bush in protest of what the administration considered the stealing of the Panamanian elections last May by Gen. Manuel Antonio Noriega. Bush sent Davis back to Panama City after the Dec. 20 invasion of Panama by U.S. forces and installation of Guillermo Endara as president. Independent observers mostly concluded Endara had won the elections by a hefty margin.

is able to accomplish a good overall accuracy on all the documents. Table 4 presents an example of the summary generated by our proposed method for a DUC 2002 document.

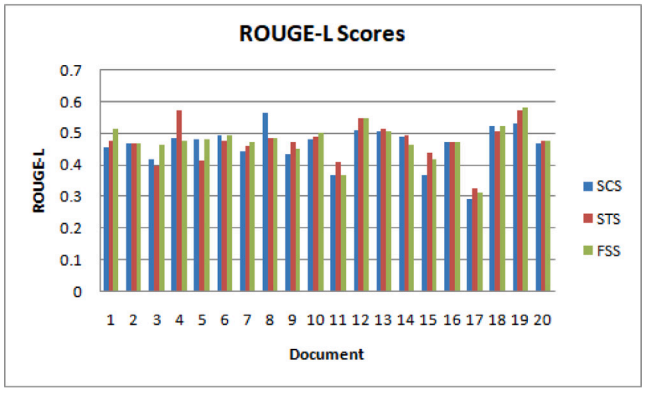
On CNN/DailyMail dataset, our method obtained the highest ROUGE-1 score and ROUGE-2 and ROUGE-L scores comparable to the best extractive summarization approaches of the literature as it can be seen in Table 6. Our algorithm achieved comparable or better ROUGE-1 score of 43.3, ROUGE-2 score of 19.0 and ROUGE-L score of 38.9. The DeepSumm method surpassed NN-SE, SummaRuNNer, Bi-AES with a very high margin of 4, 3.3, 3.4 for ROUGE-1, ROUGE-2 and ROUGE-L scores. Other state-of-the-art methods such as REFERESH and RNES that used reinforcement learning also lag in performance in comparison to our summarization proposal. We also got better ROUGE scores than those of NeuSum and HSSAS, which are based on sequence networks. DeepSumm only fall behind AREDSUM, DISCOBERT, and MATCHSUM for ROUGE-1, ROUGE-2 and ROUGE-L scores and HSG and BertSum for ROUGE-2 and ROUGE-L scores. AREDSUM, DISCOBERT, BERTSum, and MATCHSUM are intensive in terms of memory and resources. Their architectures are complex and use more number of layers as compared to our proposed architecture. As illustrated in Table 5, AREDSUM, DISCOBERT, BERTSum, and MATCHSUM utilized BERT (Devlin et al., 2019) as the base architecture of their system which has 110 Million trainable parameters with 12 GRU layers and 768 neurons in each hidden layer. Instead, our architecture consists of two bi-directional LSTM layers (one encoder and one decoder) accounting for a total of 2.5 Million trainable parameters. This states that our architecture consists of fewer parameters compared to other state-of-the-art architectures with comparatively better accuracies.



(a) ROUGE-1 graph



(b) ROUGE-2 graph



(c) ROUGE-L graph

Fig. 2. Illustration of ROUGE-1, ROUGE-2 and ROUGE-L metrics considering SCS, STS and FSS scores for the ranking of sentences on 20 randomly selected documents of DUC 2002.

We also evaluated our DeepSumm-content and DeepSumm-topic approach, which alone uses word and topic embeddings. As depicted in Table 6, DeepSumm-topic and DeepSumm-content are better than most approaches. Still, they cannot provide better ROUGE scores than when their scores are fused to capture the pertinent content in the document. The notable increase in accuracy compared to most recent approaches proved that our method is quite robust towards producing good summaries. DeepSumm can condense the salient information from the document, which is otherwise not captured alone using language models and thus, it boosts the overall accuracy of extractive summarization.

Table 5
Comparison of DeepSumm Network with other state-of-the-art Architectures.

Method	Network architecture	Neurons in hidden layer	No. of layers	Parameters (Million)
BERT	–	768	12	110
BERTSum	BERT base + 2 transformer layers	768	6	55+
MATCHSUM	2 BERTs	768	–	–
DISCOBERT	BERT as Encoder	768	–	–
AREDSUM	BERT	768	–	–
DeepSumm	**Encoder–Decoder**	**256**	**2**	**2.5**

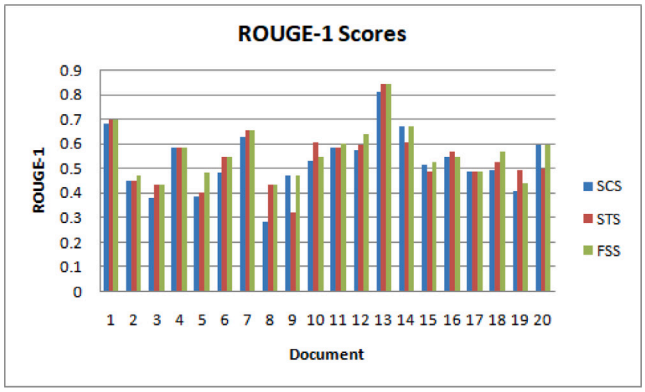
Table 6
Comparative analysis of DeepSumm with state-of-the-art algorithms on CNN/DailyMail.

Methods	ROUGE-1	ROUGE-2	ROUGE-L
NN-SE	35.5	14.7	32.2
Bi-AES	38.8	12.6	33.85
LEAD	39.2	15.7	35.5
SummaRuNNer	39.6	16.2	35.3
REFRESH	40.0	18.2	36.6
PACSUM(BERT)	40.7	17.8	36.9
RNES	41.2	18.8	37.7
STAS + PACSUM	41.26	18.18	37.48
NeuSum	41.5	19.0	37.9
DASG	41.6	18.5	37.8
JECS	41.7	18.5	37.9
HSSAS	42.3	17.8	37.6
HIBERT	42.3	19.9	38.83
PNBERT + RL	42.6	19.6	38.8
HSG + Tri-Blocking	42.9	19.7	39.2
BertSum	43.2	20.2	39.6
DeepSumm-content	41.8	18.3	37.5
DeepSumm-topic	42.9	18.8	38.3
DeepSumm	43.3	19.0	38.9
AREDSUM-CTX	43.4	20.4	39.8
DISCOBERT	43.77	20.85	40.67
MATCHSUM(RoBERTa-base)	**44.41**	**20.86**	**40.55**

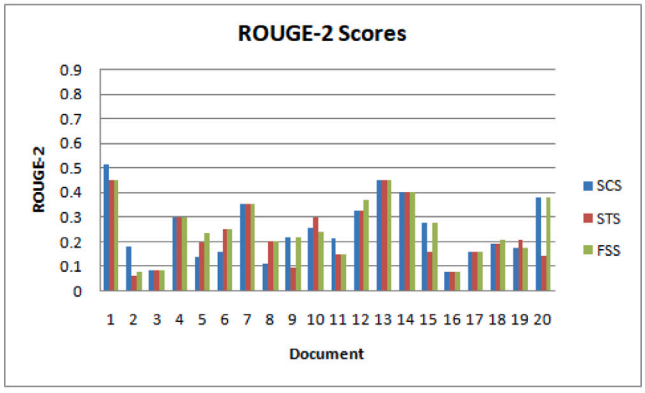
We also illustrated in Fig. 3 a comparative analysis of the performance obtained with sentence scores SCS, STS and FSS for the ranking and extraction summary sentences of documents on CNN/DailyMail dataset. Similarly to DUC 2002 dataset, it can be seen that STS yielded as good ROUGE scores on the documents as SCS did. Therefore, topic distribution sentence encodings are quite relevant for finding the pertinent content in the document to obtain semantically coherent and meaningful summaries. An exemplary summary of a CNN/DailyMail document produced by DeepSumm method is shown in Table 7.

5. Conclusions

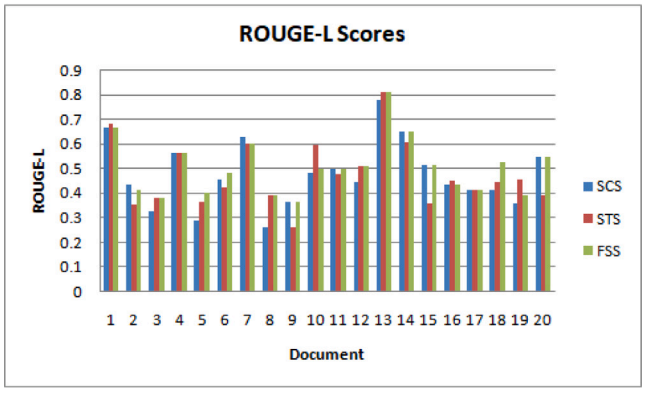
In this paper, we have presented DeepSumm, a novel method for extractive summarization which produces compact single-document representations. DeepSumm captures structural and semantic features of the document by utilizing a combination of topic and language vector encodings. We encoded the document sentences using word embeddings and word probabilistic topic distributions, creating their corresponding sentence representations. The inclusion of probabilistic topic distributions in our method makes it possible to consider the latent semantic structure of the document, which is otherwise not captured in the word embedding space. Sequence to Sequence attention networks were applied over the sentence embeddings and encodings to extract the salient sentences based on their content and topic scores, respectively. We also introduced a new novelty computation measure, SNS, to generate a non-redundant and diversified summary of the document. Last, the position of the sentence in the document was also taken into consideration using Sentence Position Score. A weighted fusion of the Sentence Content, Topic, Novelty and Position scores was used to determine the salient sentences in the document.



(a) ROUGE-1 graph



(b) ROUGE-2 graph



(c) ROUGE-L graph

Fig. 3. Illustration of ROUGE-1, ROUGE-2 and ROUGE-L metrics considering SCS, STS and FSS scores for the ranking of sentences on 20 randomly selected documents of CNN/DailyMail dataset.

The experimental results demonstrated that DeepSumm outperformed all the state-of-the-art baselines evaluated on DUC 2002 dataset,

Table 7
Gold summary and DeepSumm generated summary for a document from CNN/DailyMail dataset.

Gold summary
And this week its lyrics, hand-written in 1971 by a young folk singer called Don McLean, were sold at auction in New York for more than $ 1 million. Don McLean (pictured) is responsible American Pie, the lyrics of which have been puzzled over for decades. Argued over by generations of geeky fans, deciphered and re-deciphered by code-breaking rock nerds and considered to be poetic reflections on mid-20_{th} century U.S. social history by even groovier academics, it 's called American Pie. For more than 40 years, its lyrics have been an enigma wrapped in an eight-and-a-half minute long rock 'n' roll puzzle.
DeepSumm summary
Don Mclean pictured is responsible American pie the lyrics of which have been puzzled over for decades. Argued over by generations of geeky fans deciphered and re-deciphered by code breaking rock nerds and considered to be poetic reflections on century us social history by even groovier academics its called American pie and this week its lyrics handwritten in by a young folk singer called don mclean were sold at auction in new York for more than million. Its also a paean to education mclean loves words he says almost as much as life that may be a slight overstatement but it shows of course like all poets mclean did not give us a key to the riddle of what his song was about when he released his multi million selling single that would have spoiled it.

and achieved a competitive performance on CNN/DailyMail dataset. It has also been illustrated that high-level document features extracted using probabilistic topic distribution models are quite relevant towards generating informative summaries. There are many possibilities which can be explored in the future to extend the presented work. One possibility would be to derive other abstract features that can be combined in the existing network to increase the accuracy of the system. Secondly, we could make use of probabilistic topic distributions and sequence networks for abstractive text summarization. The third direction to investigate would be to utilize topic information in unsupervised methods for the task of extractive text summarization, which will eliminate the need for labeled summarization data for training the networks.

CRediT authorship contribution statement

Akanksha Joshi: Investigation, Conceptualization, Methodology, Software, Validation, Writing – original draft, Data curation. **Eduardo Fidalgo:** Investigation, Conceptualization, Supervision, Project administration, Resources, Writing – reviewing and editing. **Enrique Alegre:** Investigation, Conceptualization, Supervision, Writing – reviewing and editing, Funding acquisition. **Laura Fernández-Robles:** Investigation, Writing – reviewing and editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgments

This research is supported by the framework agreement between the University of León and INCIBE (Spanish National Cybersecurity Institute) under Addendum 01.

References

Al-Nabki, M. W., Fidalgo, E., Alegre, E., & Aláiz-Rodrıguez, R. (2020). File name classification approach to identify child sexual abuse. In *Proceedings of the 9th international conference on pattern recognition applications and methods, Vol. 1* (pp. 228–234).

Al-Nabki, M. W., Fidalgo, E., Alegre, E., & Fernández-Robles, L. (2020). Improving named entity recognition in noisy user-generated text with local distance neighbor feature. *Neurocomputing*, *382*, 1–11.

Al Nabki, M. W., Fidalgo, E., Alegre, E., & González-Castro, V. (2017). Detecting emerging products in TOR network based on K-shell graph decomposition. *III Jornadas Nacionales de Investigación en Ciberseguridad (JNIC)*, *1*(1), 24–30.

Al-Sabahi, K., Zu-ping, Z., & Nadher, M. (2018). A hierarchical structured self-attentive model for extractive document summarization (HSSAS). *IEEE Access*, *6*, 24205–24212.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Y. Bengio, & Y. LeCun (Eds.), *3rd international conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference track proceedings* (pp. 1–15).

Bi, K., Jha, R., Croft, B., & Celikyilmaz, A. (2021). AREDSUM: Adaptive redundancy-aware iterative sentence ranking for extractive document summarization. In *Proceedings of the 16th conference of the European chapter of the association for computational linguistics: Main volume* (pp. 281–291). Online: Association for Computational Linguistics.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77–84.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*(null), 993–1022.

Bordes, A., Chopra, S., & Weston, J. (2014). Question answering with subgraph embeddings. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 615–620). Doha, Qatar: Association for Computational Linguistics.

Cao, Z., Li, W., Li, S., Wei, F., & Li, Y. (2016). AttSum: Joint learning of focusing and summarization with neural attention. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers* (pp. 547–556). Osaka, Japan: The COLING 2016 Organizing Committee.

Carbonell, J., & Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR '98, Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval* (pp. 335–336). New York, NY, USA: Association for Computing Machinery.

Cheng, J., & Lapata, M. (2016). Neural summarization by extracting sentences and words. In *Proceedings of the 54th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 484–494). Berlin, Germany: Association for Computational Linguistics.

Chopra, S., Auli, M., & Rush, A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies* (pp. 93–98). San Diego, California: Association for Computational Linguistics.

Chung, J., Gülçehre, Ç., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555. arXiv:1412.3555.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. ArXiv abs/1810.04805.

Dieng, A. B., Wang, C., Gao, J., & Paisley, J. (2016). TopicRNN: A recurrent neural network with long-range semantic dependency. arxiv e-prints, arXiv:1611.01702.

Domínguez, V., Fidalgo, E., Biswas, R., Alegre, E., & Fernández-Robles, L. (2019). Application of extractive text summarization algorithms to speech-to-text media. In *Hybrid artificial intelligent systems* (pp. 540–550). Cham: Springer International Publishing.

Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the ACM*, *16*(2), 264–285.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, *22*(1), 457–479.

Fang, C., Mu, D., Deng, Z., & Wu, Z. (2017). Word-sentence co-ranking for automatic extractive text summarization. *Expert Systems with Applications*, *72*, 189–195.

Feng, C., Cai, F., Chen, H., & de Rijke, M. (2018). Attentive encoder-based extractive text summarization. In *CIKM '18, Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 1499–1502). New York, NY, USA: Association for Computing Machinery.

Filatova, E., & Hatzivassiloglou, V. (2004). Event-based extractive summarization. In *Text summarization branches out* (pp. 104–111). Barcelona, Spain: Association for Computational Linguistics.

Ghosh, S., Vinyals, O., Strope, B., Roy, S., Dean, T., & Heck, L. (2016). Contextual LSTM (CLSTM) models for large scale NLP tasks. ArXiv abs/1602.06291.

Hermann, K. M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. In *NIPS'15, Proceedings of the 28th international conference on neural information processing systems - Volume 1* (pp. 1693–1701). Cambridge, MA, USA: MIT Press.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *SIGIR '99*, *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 50–57). New York, NY, USA: Association for Computing Machinery.

Issam, K. A. R., Patel, S., & Subalalitha, C. N. (2021). Topic modeling based extractive text summarization. CoRR abs/2106.15313. arXiv:2106.15313.

Jadhav, A., & Rajan, V. (2018). Extractive summarization with SWAP-NET: Sentences and words from alternating pointer networks. In *Proceedings of the 56th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 142–151). Melbourne, Australia: Association for Computational Linguistics.

Jean, S., Cho, K., Memisevic, R., & Bengio, Y. (2015). On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (Volume 1: Long papers)* (pp. 1–10). Beijing, China: Association for Computational Linguistics.

Ji, Y., Cohn, T., Kong, L., Dyer, C., & Eisenstein, J. (2016). Document context language models. In Y. Bengio, & Y. LeCun (Eds.), *4rd international conference on learning representations, ICLR 2016, Puerto Rico, May 2-4, 2016, Workshop track* (pp. 1–10).

Joshi, A., Fidalgo, E., Alegre, E., & Fernández-Robles, L. (2019). SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders. *Expert Systems with Applications*, *129*, 200–215.

Khandelwal, U., He, H., Qi, P., & Jurafsky, D. (2018). Sharp nearby, fuzzy far away: How neural language models use context. In *Proceedings of the 56th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 284–294). Melbourne, Australia: Association for Computational Linguistics.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. CoRR abs/1412.6980.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, *25*, 259–284.

Lau, J. H., Baldwin, T., & Cohn, T. (2017). Topically driven neural language model. In *Proceedings of the 55th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 355–365). Vancouver, Canada: Association for Computational Linguistics.

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *ICML'14*, *Proceedings of the 31st international conference on international conference on machine learning - Volume 32* (pp. II–1188–II–1196). JMLR.org.

Li, P., Lam, W., Bing, L., Guo, W., & Li, H. (2017). Cascaded attention based unsupervised information distillation for compressive summarization. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2081–2090). Copenhagen, Denmark: Association for Computational Linguistics.

Li, P., Wang, Z., Ren, Z., Bing, L., & Lam, W. (2017). Neural rating regression with abstractive tips generation for recommendation. In *SIGIR '17*, *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 345–354). New York, NY, USA: Association for Computing Machinery.

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out* (pp. 74–81). Barcelona, Spain: Association for Computational Linguistics.

Liu, Y. (2019). Fine-tune BERT for extractive summarization. arxiv e-prints arXiv:1903.10318.

Liu, J., Hughes, D. J. D., & Yang, Y. (2021). Unsupervised extractive text summarization with distance-augmented sentence graphs. In *SIGIR '21*, *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 2313–2317). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3404835.3463111.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, *2*(2), 159–165.

McDonald, R. (2007). A study of global inference algorithms in multi-document summarization. In *ECIR'07*, *Proceedings of the 29th European conference on IR research* (pp. 557–564). Berlin, Heidelberg: Springer-Verlag.

Mehta, P., Arora, G., & Majumder, P. (2018). Attention based sentence extraction from scientific articles using pseudo-labeled data. arxiv e-prints arXiv:1802.04675.

Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing* (pp. 404–411). Barcelona, Spain: Association for Computational Linguistics.

Mikolov, T., & Zweig, G. (2012). Context dependent recurrent neural network language model. In *2012 IEEE spoken language technology workshop (SLT)* (pp. 234–239).

Nallapati, R., Zhai, F., & Zhou, B. (2017). SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI'17*, *Proceedings of the thirty-first AAAI conference on artificial intelligence* (pp. 3075–3081). AAAI Press.

Nallapati, R., Zhou, B., & Ma, M. (2017). Classify or select: Neural architectures for extractive document summarization. ArXiv abs/1611.04244.

Nallapati, R., Zhou, B., dos Santos, C., Gülçehre, Ç., & Xiang, B. (2016). Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL conference on computational natural language learning* (pp. 280–290). Berlin, Germany: Association for Computational Linguistics.

Narayan, S., Cohen, S. B., & Lapata, M. (2018a). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 1797–1807). Brussels, Belgium: Association for Computational Linguistics.

Narayan, S., Cohen, S. B., & Lapata, M. (2018b). Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human language technologies, Volume 1 (Long papers)* (pp. 1747–1759). New Orleans, Louisiana: Association for Computational Linguistics.

Narayan, S., Papasarantopoulos, N., Cohen, S. B., & Lapata, M. (2017). Neural extractive summarization with side information. arXiv e-prints arXiv:1704.04530.

Parveen, D., Ramsl, H.-M., & Strube, M. (2015). Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 1949–1954). Lisbon, Portugal: Association for Computational Linguistics.

Parveen, D., & Strube, M. (2015). Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In *IJCAI'15*, *Proceedings of the 24th international conference on artificial intelligence* (pp. 1298–1304). AAAI Press.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *ICML'13*, *Proceedings of the 30th international conference on international conference on machine learning - Volume 28* (pp. III–1310–III–1318). JMLR.org.

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543). Doha, Qatar: Association for Computational Linguistics.

Ren, P., Chen, Z., Ren, Z., Wei, F., Ma, J., & de Rijke, M. (2017). Leveraging contextual sentence relations for extractive summarization using a neural attention model. In *SIGIR '17*, *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 95–104). New York, NY, USA: Association for Computing Machinery.

Rush, A. M., Chopra, S., & Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 379–389). Lisbon, Portugal: Association for Computational Linguistics.

Shi, J., Liang, C., Hou, L., Li, J., Liu, Z., & Zhang, H. (2019). DeepChannel: Salience estimation by contrastive learning for extractive document summarization. In *The thirty-third AAAI conference on artificial intelligence, AAAI 2019, the thirty-first innovative applications of artificial intelligence conference, IAAI 2019, the ninth AAAI symposium on educational advances in artificial intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019* (pp. 6999–7006). AAAI Press.

Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In S. Dasgupta, & D. McAllester (Eds.), *Proceedings of machine learning research: vol. 28*, *Proceedings of the 30th international conference on machine learning* (pp. 1139–1147). Atlanta, Georgia, USA: PMLR.

Tang, H., Li, M., & Jin, B. (2019). A topic augmented text generation model: Joint learning of semantics and structural features. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 5090–5099). Hong Kong, China: Association for Computational Linguistics.

Tarnpradab, S., Liu, F., & Hua, K. A. (2017). Toward extractive summarization of online forum discussions via hierarchical attention networks. In V. Rus, & Z. Markov (Eds.), *Proceedings of the thirtieth international florida artificial intelligence research society conference, FLAIRS 2017, Marco Island, Florida, USA, May 22-24, 2017* (pp. 288–292). AAAI Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems 30: Annual conference on neural information processing systems 2017, 4-9 December 2017, Long Beach, CA, USA* (pp. 5998–6008).

Wan, X. (2010). Towards a unified approach to simultaneous single-document and multi-document summarizations. In *Proceedings of the 23rd international conference on computational linguistics (Coling 2010)* (pp. 1137–1145). Beijing, China: Coling 2010 Organizing Committee.

Wang, D., Liu, P., Zheng, Y., Qiu, X., & Huang, X. (2020). Heterogeneous graph neural networks for extractive document summarization. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 6209–6219). Online: Association for Computational Linguistics.

Wang, F., Orton, K., Wagenseller, P., & Xu, K. (2018). Towards understanding community interests with topic modeling. *IEEE Access*, *6*, 24660–24668.

Wang, L., Yao, J., Tao, Y., Zhong, L., Liu, W., & Du, Q. (2018). A reinforced topic-aware convolutional sequence-to-sequence model for abstractive text summarization. In *IJCAI'18*, *Proceedings of the 27th international joint conference on artificial intelligence* (pp. 4453–4460). AAAI Press.

Woodsend, K., & Lapata, M. (2012). Multiple aspect summarization using integer linear programming. In *EMNLP-CoNLL '12*, *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning* (pp. 233–243). USA: Association for Computational Linguistics.

Wu, Y., & Hu, B. (2018). Learning to extract coherent summary via deep reinforcement learning. In S. A. McIlraith, & K. Q. Weinberger (Eds.), *Proceedings of the thirty-second AAAI conference on artificial intelligence, (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018* (pp. 5602–5609). AAAI Press.

Wu, Z., Lei, L., Li, G., Huang, H., Zheng, C., Chen, E., & Xu, G. (2017). A topic modeling based approach to novel document automatic summarization. *Expert Systems with Applications*, *84*, 12–23. http://dx.doi.org/10.1016/j.eswa.2017.04.054, URL: https://www.sciencedirect.com/science/article/pii/S0957417417303020.

Xiao, W., & Carenini, G. (2019). Extractive summarization of long documents by combining global and local context. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 3011–3021). Hong Kong, China: Association for Computational Linguistics.

Xu, J., & Durrett, G. (2019). Neural extractive text summarization with syntactic compression. arXiv:1902.00863.

Xu, J., Gan, Z., Cheng, Y., & Liu, J. (2020). Discourse-aware neural extractive text summarization. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 5021–5031). Online: Association for Computational Linguistics.

Xu, S., Zhang, X., Wu, Y., Wei, F., & Zhou, M. (2020a). Unsupervised extractive summarization by pre-training hierarchical transformers. In *Findings of the association for computational linguistics: EMNLP 2020* (pp. 1784–1795). Online: Association for Computational Linguistics.

Xu, S., Zhang, X., Wu, Y., Wei, F., & Zhou, M. (2020b). Unsupervised extractive summarization by pre-training hierarchical transformers. CoRR abs/2010.08242. URL: https://arxiv.org/abs/2010.08242, arXiv:2010.08242.

Yao, K., Zhang, L., Luo, T., & Wu, Y. (2018). Deep reinforcement learning for extractive document summarization. *Neurocomputing*, *284*, 52–62.

Zhang, X., Lapata, M., Wei, F., & Zhou, M. (2018). Neural latent extractive document summarization. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 779–784). Brussels, Belgium: Association for Computational Linguistics.

Zhang, X., Wei, F., & Zhou, M. (2019). HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 5059–5069). Florence, Italy: Association for Computational Linguistics, URL: https://www.aclweb.org/anthology/P19-1499.

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *NIPS'15, Proceedings of the 28th international conference on neural information processing systems - Volume 1* (pp. 649–657). Cambridge, MA, USA: MIT Press.

Zheng, H., & Lapata, M. (2019). Sentence centrality revisited for unsupervised summarization. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 6236–6247). Florence, Italy: Association for Computational Linguistics.

Zheng, C., Zhang, K., Wang, H. J., & Fan, L. (2020). Topic-guided abstractive text summarization: a joint learning approach. CoRR abs/2010.10323. URL: https://arxiv.org/abs/2010.10323, arXiv:2010.10323.

Zhong, M., Liu, P., Chen, Y., Wang, D., Qiu, X., & Huang, X. (2020). Extractive summarization as text matching. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 6197–6208). Online: Association for Computational Linguistics.

Zhong, M., Liu, P., Wang, D., Qiu, X., & Huang, X. (2019). Searching for effective neural extractive summarization: What works and what's next. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 1049–1058). Florence, Italy: Association for Computational Linguistics, URL: https://www.aclweb.org/anthology/P19-1100.

Zhou, Q., Yang, N., Wei, F., Huang, S., Zhou, M., & Zhao, T. (2018). Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 654–663). Melbourne, Australia: Association for Computational Linguistics.

Zhou, Q., Yang, N., Wei, F., Huang, S., Zhou, M., & Zhao, T. (2020). A joint sentence scoring and selection framework for neural extractive document summarization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *28*, 671–681.