Subtopic-driven Multi-Document Summarization

Xin Zheng[1,2], **Aixin Sun**[1], **Jing Li**[3], **Karthik Muthuswamy**[2]
[1] School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2] SAP Innovation Center Network, SAP Asia Pte Ltd, Singapore
[3] Inception Institute of Artificial Intelligence Abu Dhabi, United Arab Emirates
xzheng008@e.ntu.edu.sg, axsun@ntu.edu.sg,
jingli.phd@hotmail.com, karthik.muthuswamy@sap.com

Abstract

In multi-document summarization, a set of documents to be summarized is assumed to be on the same topic, known as the *underlying topic* in this paper. That is, the underlying topic can be collectively represented by all the documents in the set. Meanwhile, different documents may cover various different subtopics and the same subtopic can be across several documents. Inspired by topic model, the underlying topic of a document set can also be viewed as a collection of different subtopics of different importance. In this paper, we propose a summarization model called *STDS*. The model generates the underlying topic representation from both document view and subtopic view in parallel. The learning objective is to minimize the distance between the representations learned from the two views. The contextual information is encoded through a hierarchical RNN architecture. Sentence salience is estimated in a hierarchical way with subtopic salience and relative sentence salience, by considering the contextual information. Top ranked sentences are then extracted as a summary. Note that the notion of subtopic enables us to bring in additional information (*e.g.,* comments to news articles) that is helpful for document summarization. Experimental results show that the proposed solution outperforms state-of-the-art methods on benchmark datasets.

1 Introduction

Multi-document summarization (MDS) is useful in many applications, *e.g.,* summarizing answers in forums (Song et al., 2017) and reports of burst events (Kedzie et al., 2016). In MDS, the documents in a set are assumed to share the same underlying topic. Given a set of documents to be summarized, the important information is collectively determined by all the documents in the set, rather than simple integration of key points in each document. Hence, the correlations among documents become crucial for identifying the important information to be included in a summary. Moreover, each sentence should not be interpreted independently. Its context (or the surrounding sentences) does affect the information expressed in a sentence (Nenkova et al., 2006; Ren et al., 2017). Therefore, both correlations among documents and contextual information within a single document should be considered in MDS.

Various methods have been developed for MDS. Graph-based models (Mihalcea and Tarau, 2004; Erkan and Radev, 2004) blend sentences from different documents together and attempt to leverage the correlations among documents to extract the most representative sentences. Nonetheless these methods are less effective when the sentence graph is not well connected. Thus, many studies attempt to construct well connected graphs, *e.g.,* group sentences into clusters (Wan and Yang, 2008; Banerjee et al., 2015) and construct dense features with distributed embeddings (Yasunaga et al., 2017). However, these methods do not address the context information for sentences in each individual document. There are also models exploring fine-grained word or phrase level information (Li et al., 2015, 2017a,b). The resultant summary consists of salient words or phrases that are selected by integer linear programming (ILP). Recently, many studies have taken advantage of the strengths of neural models (Ren et al., 2017; Zhou et al., 2018; Lebanoff et al., 2018). However, these models are not designed to well deal with both correlations among documents in the set and the contextual information in each individual document.

To address the two aforementioned issues, we propose a subtopic-driven summarization solution, named *STDS*. To encode sequential context information, we adopt a hierarchical bidirectional RNN to produce representations for sentences,

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3144–3153,
Hong Kong, China, November 3–7, 2019.

documents, and the underlying topic. As stated earlier, within a document set, the documents may cover various subtopics and the same subtopic can be across several articles. These subtopics implicitly reflect the correlations among documents. Inspired by topic model (Blei et al., 2003), we simply assume that k latent subtopics are depicted by the documents. We design our model to learn the representations of "subtopics" by itself, by assigning sentences to different subtopics in a soft manner instead of hard clustering. Thus, the underlying topic of a given document set can be presented from document view and subtopic view, in parallel. The subtopic view also gives us the flexibility of incorporating additional information (*e.g.,* comments to news articles). Readers' comments have been found useful for highlighting crucial information (Li et al., 2017a). The salience of subtopics are estimated by using attention mechanism (Bahdanau et al., 2015) with respect to the underlying topic generated from the document view. Similarly, relative sentence salience can be estimated for each subtopic by considering context information. By multiplying the saliences of subtopics and sentences, an overall ranking of sentences can be obtained. Top-ranked sentences are extracted as a summary within a given length limit. The contributions of this work are as follows:

- We propose to explore correlations among documents with "subtopics". The subtopics and documents provide us parallel views of the underlying topic and enable sentence and subtopic salience estimation in an unsupervised fashion.
- We build up a unified model that tackles the correlations among documents and contextual information within each document together in an inherent way.
- Our model can be applied to documents with or without comments. Extensive experiments are conducted on benchmark datasets, *i.e.,* RA-MDS and DUC 2004. The experimental results show that our STDS outperforms state-of-the-art baselines.

2 Related Work

2.1 Extractive Methods

Extractive methods select sentences from documents to form a summary. A typical framework is based on a graph, where sentences are vertices and similarities between sentences are edge weights, *e.g.,* TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004). They apply a random walk to explore the relationships among sentences and then produce a sentence ranking. MEAD (Radev et al., 2000) is a centroid-based method, which ranks sentences based on a set of features, including centroid value, positional value, first sentence overlap and redundancy.

However, the graph-based models do not perform well when the sentence graph is not well connected. Wan and Yang (2008) claim that the document set is usually composed of a few themes, which are represented by sets of sentences. They apply conditional Markov Random Walk and HITS on clusters, separately. Alternatively, Haghighi and Vanderwende (2009) adopt a hierarchical Latent Dirichlet Allocation based model to discover the multiple themes within a document set. Similarly, Gong *et al.* (2010) use the theme structure to define the representation for each sentence. However, their solutions only consider statistical knowledge. Semantic and contextual information among sentences are neglected.

Following the idea of themes, Banerjee *et al.* (2015) suggest that the sentences in the most important document of the set are relevant to the sentences in the other documents. Hence, they cluster sentences based on those in the most important document. However, there may not always exist a most important document in the set. Further, the relationships among documents are not necessarily to be conclusive. In fact, there are many kinds of relationships among documents (*e.g.,* similar, complementary, or evolutionary).

Liu *et al.* (2015) adopt the idea of reconstruction. They apply a two-level sparse representation model and reconstruct the document by extracted sentences with constraints. Similarly, Ma *et al.* (2016) try to minimize the reconstruction error between selected sentences and the document set with a neural model. Cao *et al.* (2017) make use of multi-task learning by incorporating classification task with summarization to train better sentence representations for reconstruction.

2.2 Abstractive Methods

Summarizing multiple documents in an abstractive way is even harder. The generation should take many inputs into account. Some abstrac-

tive methods on single document summarization have been proposed recently (See et al., 2017; Zhou et al., 2018), and they benefit from an attentive sequence-to-sequence model (Sutskever et al., 2014). In fact, abstractive approaches for MDS are more like words or phrases recombination.

Bing *et al.* (2015) propose an abstractive MDS solution. They extract salient noun and verb phrases from a constituency tree, then produce sentences with representative phrases via integer linear programming. Later, Li *et al.* (2017b; 2017c) adopt a similar two-stage model, but they first estimate sentence and phrase salience via an auto-encoder framework.

Recently, some studies have turned to readers' comments to help identify crucial points to include in a summary. Li *et al.* (2015) propose a sparse coding method to generate summaries that not only cover key content in news but also focuses highlighted by readers' comments. However, they do not consider semantic information. Later, they propose a deep learning method (Li et al., 2017a) that jointly models the focuses of news set and readers' comments. But they do not tackle sequential context information among sentences and treat them as separate instances. In contrast, we deal with sequential context information within each document and the relationships among documents.

3 Preliminary

We define our problem as follows. Given a set of news documents on the same topic, $D = \{d_1, d_2, ..., d_n\}$, we also have a set of comments $C_i = \{c_{i,1}, c_{i,2}, ..., c_{i,h}\}$ for each document d_i. Both news d_i and comment $c_{i,j}$ consist of a sequence of sentences, *i.e.*, $d_i = \{s_1, s_2, ..., s_{|d_i|}\}$ and $c_{i,j} = \{y_1, y_2, ..., y_{|c_{i,j}|}\}$. We aim to extract representative *news sentences* to summarize the set of news articles by considering both news and comments.

To better leverage comments, we explore the correlation between news and comments based on the dataset provided by Li *et al.* (2017a). The dataset contains 45 topics (*i.e.,* sets of news articles), and each topic includes 10 news articles. The number of comments associated with each news article is not evenly distributed and ranges from 0 to 248. Overall, on average, each topic is associated with 215 comments and 940 comment sentences.

The correlations between news and comments determines how we use them. Thus, we make the following analysis. Since the news documents share the same topic, comments about one news may also be related to the others. Meanwhile, it is common for users to make comments after reading through several topic-related news articles. Thus, comments for one news could also contain information from related articles. To verify our assumption, we calculate the ratio of overlapped vocabulary for each news and its comments, over news vocabulary, and the average is 0.345. Against news-comments pairs, we compute the ratio of overlapped vocabulary for all the news and comments of the same topic, over news vocabulary, and the average is 0.447. Thus, considering comments as related to the whole news set is more reasonable than news-comment pairs, which coincides with (Li et al., 2015, 2017a).

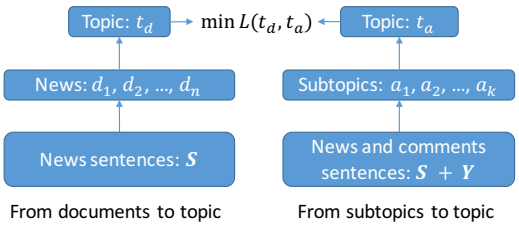

Figure 1: Framework of STDS. The underlying topic of the news set is represented from a document view and a subtopic view respectively. The target is to minimize the distance between the two representations $L(t_d, t_a)$.

In short, our model inputs are a news set $D = \{d_1, d_2, ..., d_n\}$ with news sentences S, and a set of comments $C = \{c_1, c_2, ..., c_m\}$ with comment sentences Y. Later, we will use boldface of each notation to represent its embedding.

4 Model Description

Our model tackles correlations among documents and sequential context in each individual document, through "subtopics" and hierarchical semantic embedding. We first introduce the embedding generation with context for sentences, documents, and subtopics in Sections 4.1 and 4.2. Then, we describe salience estimation for subtopics and sentences in Sections 4.3 and 4.4.

4.1 Sentence and Document Representation

We start representation generation from words, and pre-train word embedding $\boldsymbol{W} \in \mathbb{R}^{|\mathcal{V}| \times u}$ by Word2Vec (Mikolov et al., 2013) on both news and comments, where $\mathcal{V}$ is the overall vocabulary.

Note that we use u to represent the vector dimension throughout the model description (specific value settings are given in Section 5). The representations for news sentences $\boldsymbol{S}$ and news documents $\boldsymbol{D}$ are learned in a hierarchical structure, as shown in Figure 2. Specifically, the embeddings of words $\boldsymbol{w}$ in each sentence are fed into a bidirectional RNN encoder-decoder framework (Cho et al., 2014), which inherently encodes word sequence information in sentence representations. GRU (Chung et al., 2014) is adopted as the basic RNN unit. Each sentence acts as both encoder input and decoder target. Thus, the encoder-decoder could be considered as an RNN auto-encoder. The concatenation of hidden vectors in both directions at the last step of the encoder is adopted as the sentence embedding, $\boldsymbol{s} = [\overrightarrow{\boldsymbol{h}_{|s|}}^\top; \overleftarrow{\boldsymbol{h}_{|s|}}^\top]^\top$. The comment sentence embedding $\boldsymbol{Y}$ is generated in the same way. The loss function for the RNN encoder-decoder is:

$$\mathcal{L}_e = -\sum_{X \in S \cup Y} \log p(X'|X;\theta) \tag{1}$$

where X' is the predicted result and θ denotes the parameters.

Then, news sentence embeddings $\boldsymbol{s}$ are fed into the next level bidirectional RNN in order, as shown in Figure 2, to generate a document representation $\boldsymbol{D} \in \mathbb{R}^{|D| \times u}$. Note that no decoding process exists from sentence to document. At each encoding step for document embedding, we obtain the hidden vectors on both directions for each sentence, which contain contextual information along with sentences. We concatenate them as a context-enriched sentence embedding $\tilde{\boldsymbol{S}} \in \mathbb{R}^{|S| \times u}$, where $\tilde{\boldsymbol{s}}_i = [\overrightarrow{\boldsymbol{h}_i}^\top; \overleftarrow{\boldsymbol{h}_i}^\top]^\top$. Again, we take the concatenation of hidden vectors in both directions at the last step as the document embedding $\boldsymbol{d} = [\overrightarrow{\boldsymbol{h}_{|d|}}^\top; \overleftarrow{\boldsymbol{h}_{|d|}}^\top]^\top$.

For now, the sequential contextual information in an individual document is encoded in the context-enriched sentence embedding $\tilde{\boldsymbol{S}}$ with the hierarchical bidirectional RNN inherently.

4.2 Subtopic Representation

As stated in Section 1, we assume a news set contains k latent subtopics, which are expected to be learned automatically from sentences of news and comments. This is because readers' comments are found useful to highlight important information in news (Li et al., 2015, 2017a).

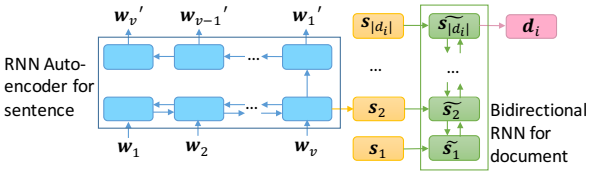

Figure 2: Hierarchical RNN structure for sentence representation $\boldsymbol{s}_j$ and document representation $\boldsymbol{d}_i$ construction. $\boldsymbol{w}_i$ is the embedding of words in each sentence and $\boldsymbol{s}_j$ is the concatenation of hidden vectors in both directions at the last step of the sentence encoder. $\tilde{\boldsymbol{s}}_j$ is a context-enriched sentence embedding, which is the concatenation of hidden states at each step of the document encoder.

Recall that we construct a document embedding from news sentence representations $\boldsymbol{S}$, which are generated from a word sequence. For subtopic representation, we also utilize $\boldsymbol{S}$, and incorporate the comment sentence embedding $\boldsymbol{Y}$. That is, we only take a single sentence without context information to build subtopic representations, as in a typical topic model. Then, a soft clustering is employed, so that sentences of news and comments $\boldsymbol{E} = [\boldsymbol{S}^\top; \boldsymbol{Y}^\top]^\top \in \mathbb{R}^{(|S|+|Y|) \times u}$ can be partial membership of multiple subtopics. Next, a nonlinear transformation $tanh$ is utilized to fuse clustered sentence information into subtopic representations $\boldsymbol{A} \in \mathbb{R}^{k \times u}$ as follows:

$$\boldsymbol{A} = tanh((\boldsymbol{H}\boldsymbol{E} + \boldsymbol{b}_h)\boldsymbol{U} + \boldsymbol{b}_u) \tag{2}$$

where $\boldsymbol{H}, \boldsymbol{U}, \boldsymbol{b}_h, \boldsymbol{b}_u$ are trainable parameters.

We hope that each subtopic embedding can represent a unique facet of the underlying topic. That means the overlap across different subtopics should be as small as possible. Inspired by Luxburg (2007), we constrain the subtopic embedding to be orthogonal with each other:

$$\mathcal{L}_r = \|\boldsymbol{A}\boldsymbol{A}^\top - \boldsymbol{I}_{k \times k}\|. \tag{3}$$

4.3 Salience Estimation for Subtopic

The purpose of introducing subtopics is not to make the summary diverse. Instead, we aim to identify the important information (and relieve distractions from less crucial content) in the news set, through the notion of subtopics. This is achieved by subtopic salience estimation. We consider the underlying topic of a news set from two parallel perspectives, as shown in Figure 1: (i) the explicit composition of documents, and (ii) the implicit constitution of subtopics. Therefore, from either document view or subtopic view, we should

be able to obtain similar representations for the underlying topic of the news set.

From documents to the underlying topic. We obtain a document representation $\boldsymbol{d}$ with a hierarchical RNN. Again, we adopt a bidirectional RNN to encode document embeddings to a topic representation $\boldsymbol{t}_d$, which is the concatenation of hidden vectors in both directions at the last step. Here, the document sequences do not significantly affect model performance in our experiments. In fact, one can also try to use a CNN to get $\boldsymbol{t}_d$. However, our experiments suggest RNN performs better than CNN.

From subtopics to the underlying topic. As we stated before, not all subtopics are equally important to the underlying topic. We estimate subtopic salience in order to focus on key information *i.e.,* to generate a distilled topic representation $\boldsymbol{t}_a$. This is different from the topic vector $\boldsymbol{t}_d$ constructed from documents, which encodes all information from all documents.

To estimate subtopic salience, we apply an attention mechanism (Bahdanau et al., 2015) on each subtopic representation $\boldsymbol{a}_i$ with respect to the topic representation $\boldsymbol{t}_d$, which encodes comprehensive information from documents:

$$\alpha_i = \frac{\exp(e_i)}{\sum_{k=1}^{m}\exp(e_k)}, e_i = \boldsymbol{v}^\top tanh(\boldsymbol{H}\boldsymbol{a}_i + \boldsymbol{U}\boldsymbol{t}_d) \tag{4}$$

where $\boldsymbol{H}$, $\boldsymbol{U}$ and $\boldsymbol{v}$ are trainable parameters. We feed the weighted subtopic embedding $\alpha_i\boldsymbol{a}_i$ into a bidirectional RNN. The reason for using an RNN is the same as that for t_d generation, mentioned above. The concatenation of hidden vectors in both directions at the last step is taken as the topic representation $\boldsymbol{t}_a$, generated from subtopics. Both topic representations $\boldsymbol{t}_a$ and $\boldsymbol{t}_d$ should be similar to each other, because they denote the same underlying topic. Since the two vectors are from identical structure as Siamese network (Koch et al., 2015), we adopt contrastive loss (Hadsell et al., 2006) to measure the distance between them:

$$\mathcal{L}_t = \beta * (max\{0, 1 - \boldsymbol{t}_d^\top \boldsymbol{t}_a\})^2 + (1-\beta) * (\boldsymbol{t}_d - \boldsymbol{t}_a)^2 \tag{5}$$

where β is a hyper-parameter, empirically set to 0.5.

4.4 Salience Estimation for Sentence

Note that, each sentence is also of different importance with respect to different subtopics, which is called *relative sentence salience*. Recall that we claim the sequential context information affects sentence salience. Thus, we adopt the context-enriched sentence embedding $\tilde{\boldsymbol{s}}_j$, (which is presented in Section 4.1), to estimate relative sentence salience $\gamma_{i,j}$ for each subtopic $\boldsymbol{a}_i$, similar as Equation 4.

The subtopic representations are generated from both news $\boldsymbol{S}$ and comments $\boldsymbol{Y}$. On the other hand, the subtopics represent information conveyed by the news set. Thus, we should also be able to construct subtopic vectors solely based on news sentences. To avoid diminishing of comments $\boldsymbol{Y}$, we use the context-enriched news sentence embedding $\tilde{\boldsymbol{S}}$ to approximate subtopic representations $\boldsymbol{A}$, similar as Equation 2:

$$\boldsymbol{A}_{\tilde{s}} = tanh(\boldsymbol{\gamma}\tilde{\boldsymbol{S}}\boldsymbol{H} + \boldsymbol{b_h}) \tag{6}$$

where $\boldsymbol{H}$ and $\boldsymbol{b_h}$ are trainable parameters. $\boldsymbol{A}$ and $\boldsymbol{A}_{\tilde{s}}$ should be similar and we measure the distance between them with contrastive loss:

$$\mathcal{L}_a = \beta * (max\{0, \|\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}_{\tilde{s}}^\top\|\})^2 + (1-\beta) * (\boldsymbol{A} - \boldsymbol{A}_{\tilde{s}})^2 \tag{7}$$

where β is a hyper-parameter, empirically set to 0.5. With backpropagation, relative sentence salience scores $\boldsymbol{\gamma}$ can be learned.

The overall training objective becomes:

$$\mathcal{L} = \mathcal{L}_e + \mathcal{L}_r + \mathcal{L}_t + \mathcal{L}_a. \tag{8}$$

To obtain the global salience score for each sentence, we multiply the max relative salience of each sentence over subtopics $\max \boldsymbol{\gamma}_{:,j}$, with the corresponding subtopic salience α_i, (*i.e.,* $i = \arg\max \boldsymbol{\gamma}_{:,j}$). Top ranked sentences are extracted as a summary with length limited (*i.e.,* 100 words).

5 Experiments

5.1 Datasets and Experimental Settings

Our model is proposed to incorporate comments for summarization, but it can also be applied when no comments are available. Hence, we evaluate our model on two benchmark datasets: one includes comments and the other does not.

RA-MDS (Li et al., 2017a): This dataset includes comments and is the one we conduct analysis on in Section 3. For each topic, 4 reference summaries within 100 words are generated by human. The average news length is 27 sentences, and each sentence contains 25 words on average. To alleviate comment bias when generating subtopics, we randomly sample 50 comments

for each topic, and repeat the sampling 10 times. Thus, we obtain 450 news document sets. At test stage, for each topic, we randomly select 50 comments, together with the news set for evaluation.

DUC2004[1]: The data is from Task 2 of DUC 2004. It has 50 topics, each of which consists of 10 news articles. For each topic, multiple reference summaries generated by human judges are provided.

Experimental settings. The notation u in Section 4 generally represents the hidden vector dimension. We set the word embedding dimension as 300. The hidden units for all GRU cells in one direction are 128, so the hidden vector dimension after concatenating both directions is 256. The dimensions of the linear operations in Equations 2 and 6 are also set to 256. We perform sensitivity experiments on the subtopic number and empirically set the number k to 5. The word embeddings are pre-trained on two datasets separately and are immutable during the training process.

The batch size is 1, which is the data from one news set. One iteration is training one batch of data. We adopt Adam (Kingma and Ba, 2015) as the optimizer. The learning rate is 0.001 and the value is decayed by 0.96 after 1,000 iterations. To reduce the vanishing and exploding gradient problems for RNN training, we apply gradient norm clipping strategy (Pascanu et al., 2013) and set the bounded norm as 1. We stop training when the loss stops decreasing after more than 3 iterations. Our model is implemented with TensorFlow version 1.3[2] and runs on a single GPU.[3]

5.2 Baseline Models

We compare our model with both traditional methods and state-of-the-art models on both datasets.

Lead: For news, the first several sentences usually act as good summarization. The news sentences are ordered chronologically and top ones are extracted until meeting the length limit.

MEAD (Radev et al., 2000): It detects the topics of given document set and uses information from the centroid of each topic to select sentences.

LexRank (Erkan and Radev, 2004), **TextRank** (Mihalcea and Tarau, 2004): Both are unsupervised graph-based models. They perform PageRank on a sentence graph, where sentence similarities are edge weights.

APSM (Bing et al., 2015): It extracts noun and verb phrases from a contiguity tree and estimates the phrase salience based on handcrafted features. Then, an integer linear programming process is employed to optimize the phrase selection.

RA-Sparse (Li et al., 2015): This model tackles reader-aware MDS problem. A sparse-coding-based method is used to calculate sentence salience by jointly considering news documents and readers' comments.

RAVAESum, RAVAESum/NC (Li et al., 2017a): They explore the relationships between news and comments via representation matrix multiplication, from which a weight matrix is obtained. The magnitudes of the weight matrix are taken as sentence salience. Then, integer linear programming is used to optimize the summary construction. RAVAESum/NC removes readers' comments from the input and trains the model solely based on news.

The methods below are baselines used on DUC 2004 dataset without comments.

CLASSY04 (Conroy et al., 2004): It performs the best on the official DUC 2004 evaluation. It applies a Hidden Markov Model and uses topic signature as feature.

GreedyKL (Haghighi and Vanderwende, 2009): It focuses on words distribution, and aims to minimize the KL divergence between word probability distribution estimated from summary and that from the input using a greedy approach.

RegSum (Hong et al., 2014): This is a supervised method. It adopts weights estimated from three unsupervised methods with a set of handcrafted features for salience estimation.

ILPSumm (Banerjee et al., 2015): It groups similar sentences and generates K-shortest paths from sentences in each cluster using a word-graph structure. Then, it selects sentences from the set of shortest paths via ILP.

AAPRW (Wang et al., 2017): This is an adjustable affinity-preserving random walk model to keep the summary diverse.

GRU-GCN (Yasunaga et al., 2017): It employs an RNN to obtain sentence embedding and feeds them to a Graph Convolutional Network (GCN) as node features. High-level features are generated from the GCN for sentence salience estimation.

CRSum,CRSum-SF (Ren et al., 2017): CRSum

[1] https://www-nlpir.nist.gov/projects/duc/data.html

[2] https://www.tensorflow.org/versions/r1.3/api_docs/python/

[3] Tesla P100, 3,584 Cuda cores, 16G GPU memory.

Method	R-1	R-2	R-SU4
Lead	0.384	0.110	0.144
MEAD	0.402	0.141	0.171
TextRank	0.402	0.122	0.159
LexRank	0.425	0.135	0.165
APSM	0.422	0.157	0.188
RA-Sparse	0.442	0.157	0.188
RAVAESum/NC	0.437	0.162	0.189
RAVAESum	<u>0.443</u>	<u>0.171</u>	<u>0.196</u>
STDS/NC	0.443	0.163	0.177
STDS	**0.456**	**0.187**	**0.205**

Table 1: Full-length ROUGE $\boldsymbol{F_1}$ score of the proposed methods and baselines on RA-MDS dataset. "R" indicates ROUGE. All the ROUGE scores reported in this work are with significance test and the values fall in a 95% confidence interval with at most ± 0.25 of reported results calculated by the official ROUGE script with default settings. The best $\boldsymbol{F_1}$ scores are in bold and second best are underlined.

Method		R-1	R-2
Supervised Method	CLASSY04	0.376	0.090
	RegSum	0.386	0.098
	GRU-GCN	0.382	0.095
	PG-MMR	0.364	0.094
	CRSum	0.382	0.097
	CRSum-SF	<u>0.395</u>	0.106
Unsupervised Method	LexRank	0.359	0.075
	GreedyKL	0.379	0.085
	AAPRW	0.389	0.101
	ILPSum	0.392	**0.119**
	STDS/NC	**0.397**	<u>0.107</u>

Table 2: ROUGE ***Recall*** for the proposed method and baselines on DUC 2004 dataset. Best ***Recall*** scores are in bold and second best are underlined.

exploits context features within sentences and among sentences with a two-level attention mechanism. CRSum-SF refers to the model combining surface features.

PG-MMR (Lebanoff et al., 2018): It leverages the Maximal Marginal Relevance method to select representative sentences from input documents avoiding duplicates, and uses an existing abstractive encoder-decoder model (See et al., 2017) to generate an abstractive summary.

Our Model STDS: We feed the first 25 words of each sentence and the first 27 sentences of each document into our model (*i.e.,* which are the average numbers of words and sentences of RA-MDS dataset). Experimental results suggest no significant difference when feeding all the data. Since STDS requires readers' comments, we only evaluate it on RA-MDS dataset.

Our Model STDS/NC: This is a variant of STDS without readers' comments as input. The subtopic embeddings are generated solely based on news sentence embedding $\boldsymbol{s}$. For DUC 2004, the first 20 words of each sentence and the first 20 sentences of each document are fed into the model. The other settings remain the same. When no readers' comments exists, our model can be applied in this manner and we test STDS/NC on both datasets.

5.3 Performance Comparison

We evaluate the models by pyrouge[4], which is a python wrapper of the official ROUGE toolkit (Lin, 2004). We limit the summary length to 100 words on both datasets to compare with baselines. Following Li *et al.* (2017a), we report ROUGE F_1 on RA-MDS in Table 1. For DUC 2004, we report ROUGE $Recall$ following previous studies (Wang et al., 2017) in Table 2.

RA-MDS: From Table 1, we observe that our STDS outperforms all baselines on all of ROURGE-1, ROUGE-2 and ROUGE-SU4, which demonstrates the superiority of our model. By incorporating readers' comments, STDS and RAVAESum achieve better results than STDS/NC and RAVAESum/NC. Thus, readers' comments do help construct better summaries. The variant of our model, STDS/NC, achieves comparable performance with the best baseline, RAVAESum, on ROUGE-1. This shows the subtopics and contextually enriched semantic embedding are important for summarization.

MEAD, LexRank and TextRank all tackle correlations among documents, but they do not consider surrounding context influence. STDS and STDS/NC take both elements into consideration and perform much better than these baselines. Thus, context information contributes to better summaries. If this comparison is not direct enough because our models STDS and STDS/NC can benefit from neural model, we compare STDS and RAVAESum next. Both RAVAESum and STDS are neural models. The difference is that our STDS does not take all news and comment sentences as a whole set. STDS distributes sentences into different subtopics of varying importance and incorporates surrounding sentences as context to produce sentence representations. Our model performs better than RAVAESum. Therefore, both

[4] https://pypi.python.org/pypi/pyrouge/0.1.0

[Extracted] d2s1t1: Facebook launches Internet.org app to let users access basic Internet services for free. **d6s2t1:** Facebook's Internet.org project is taking another step toward its goal of bringing the Internet to people who are not yet online, launching an app Thursday in Zambia. **d3s4t2:** Facebook will not pay Airtel for the bandwidth, Rosen said, but Airtel will benefit as users who are exposed to Internet services eventually decide to pay for broader, unrestricted access. **d8s3t1:** The Internet.org app will give subscribers of Zambia's Airtel phone company access to a set of basic internet services for free.
[Reference] r1: Facebook launched an Internet.org app in Zambia to provide free access to Facebook and other online services including Wikipedia, Google Search, AccuWeather and websites offering health and other services. **r2:** Facebook aims to bring the Internet to people who are not yet online. **r3:** The new app has the potential to boost the size of Facebooks audience, which currently totals 1.32 billion monthly users. **r4:** Facebook will not pay Airtel for the bandwidth, but Airtel will benefit as users may decide to pay for unrestricted access. **r5:** Google has run its own zero-data initiative called Free Zone including a partnership with Airtel in India.

Table 3: Summary example.

Subtopic	Salience	Top-ranked sentence of the subtopic
t1	0.312	Facebook launches Internet.org app to let users access basic Internet services for free.
t2	0.284	Facebook will not pay Airtel for the bandwidth, Rosen said, but Airtel will benefit as users who are exposed to Internet services eventually decide to pay for broader, unrestricted access.
t3	0.197	Online services accessible through the app range from AccuWeather to Google search, Wikipedia, a job search site as well as a breadth of health information.
t4	0.122	Mark Zuckerberg has said that making connectivity affordable and convincing people that the Internet is something they need are bigger hurdles to connecting people than "satellites or balloons."
t5	0.085	The app works on Android phones as well as the simple "feature phones" that are used by the majority of people in Zambia, said Guy Rosen, product management director at Internet.org.

Table 4: Subtopic example.

subtopics and context information contribute to STDS's good performance. Since our model is unified, (*i.e.,* sentence and subtopic salience estimation relies on both subtopics and contextual information), we cannot carry out ablation experiments to verify the effectiveness of the two factors individually.

DUC 2004: Note that our model STDS is designed to incorporate readers' comments. When no comments are available, STDS/NC performs fairly well, as shown in Table 2. We observe that STDS/NC achieves comparable results as state-of-the-art unsupervised method ILPSum and supervised method CRSum-SF, and outperforms the other strong baselines. Both GRU-GCN and STDS/NC exploit the correlations among documents, but STDS/NC encodes contextual information into the sentence embedding. The better performance of STDS/NC than GRU-GCN indicates the effectiveness of surrounding context. Moreover, both CRSum and STDS/NC encode contextual information into sentence embedding, and STDS/NC outperforms the supervised CRSum. Thus, this suggests that our model benefits from the subtopics with different saliences. They provide a better understanding of the correlations among documents and lead to the important subtopics. With the comparisons, we demonstrate the effectiveness of both factors: subtopics and context information.

In a word, without training using reference summaries, STDS and STDS/NC outperform most baseline methods and achieve comparable results with state-of-the-art methods. This shows that our model can be applied to common situations without readers' comments.

5.4 Case Study with Example Summary

We present one summary example in Table 3. The notations before extracted sentences represent **d**ocument, **s**entence, and sub**t**opic respectively. The number following **d** is the document number, labeled in the raw dataset. The number after **s** is the sentence position in the document. And the number following **t** indicates the subtopic ranking position based on subtopic salience, which is the same as in Table 4.

As observed, the extracted summary aligns to content of **r1**, **r2** and **r4** in reference summary. After checking the raw documents, we find few content is related to reference sentences **r3** and **r5**.

Table 4 shows the identified subtopic information of the same document set as for Table 3. We list the salience scores and top-ranked sentences to better understand the subtopics. We consider that the top-ranked sentence regarding each subtopic represents the main content of the subtopic. In this case, the subtopics do capture different aspects of the document set. And they provide a structured way to understand the set of documents. The salience scores demonstrate that the subtopics

are not equally important. The salience estimation of subtopics aims to facilitate the summarization process focusing on important information and reduce the distraction from trivial content. As in Table 3, the extracted summary sentences only cover two most important subtopics and they come from different documents. Thus, many overlaps exist among different documents, which complement each other and form the final subtopics identified by our model.

However, redundancy occurs in the extracted summary, *e.g.,* both **d2s1t1** and **d8s3t1** mention "Internet.org app provide Internet services for free". Besides, incoherence also exists, *e.g.,* **d8s3t1** should be in front of **d3s4t2**. This is the inherent drawback of extractive summarization. Thus, there is still great room for improvement. We will deal with these issues in our future work.

6 Conclusion

In this paper, we propose an unsupervised MDS solution, which deals with both sequential context information in a single document and the correlations among documents. Hierarchical RNN is adopted to encode context information in sentence and document embeddings. We transform correlations among documents into subtopics expressed by sentences in different documents. Readers' comments are incorporated into subtopics formation to highlight important information in news. The subtopics provide us another angle to understand the underlying topic besides the document view. The sentence salience is estimated with subtopic salience and relative sentence salience in the hierarchical way as well. Extensive experiments show that the proposed model outperforms state-of-the-art baselines.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. 2015. Multi-document abstractive summarization using ILP based multi-sentence compression. In *IJCAI*, pages 1208–1214.

Lidong Bing, Piji Li, Yi Liao, Wai Lam, Weiwei Guo, and Rebecca J. Passonneau. 2015. Abstractive multi-document summarization via phrase selection and merging. In *ACL, Vol. 1*, pages 1587–1597.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *JMLR*, 3:993–1022.

Ziqiang Cao, Wenjie Li, Sujian Li, and Furu Wei. 2017. Improving multi-document summarization via text classification. In *AAAI*, pages 3053–3059.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*, pages 1724–1734.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555.

John M Conroy, Judith D Schlesinger, Jade Goldstein, and Dianne P Oleary. 2004. Left-brain/right-brain multi-document summarization. In *DUC 2004*.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *JAIR*, 22:457–479.

Shu Gong, Youli Qu, and Shengfeng Tian. 2010. Subtopic-based multi-documents summarization. In *CSO*, volume 2, pages 382–386. IEEE.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *HLT-NAACL*, pages 362–370.

Kai Hong, John M. Conroy, Benoît Favre, Alex Kulesza, Hui Lin, and Ani Nenkova. 2014. A repository of state of the art and competitive baseline summaries for generic news summarization. In *LREC*, pages 1608–1616.

Chris Kedzie, Fernando Diaz, and Kathleen R. McKeown. 2016. Real-time web scale event summarization using sequential decision making. In *IJCAI*, pages 3754–3760.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2.

Logan Lebanoff, Kaiqiang Song, and Fei Liu. 2018. Adapting the neural encoder-decoder framework from single to multi-document summarization. In *EMNLP*, pages 4131–4141.

Piji Li, Lidong Bing, and Wai Lam. 2017a. Reader-aware multi-document summarization: An enhanced model and the first dataset. In *NFiS@EMNLP*, pages 91–99.

Piji Li, Lidong Bing, Wai Lam, Hang Li, and Yi Liao. 2015. Reader-aware multi-document summarization via sparse coding. In *IJCAI*, pages 1270–1276.

Piji Li, Wai Lam, Lidong Bing, Weiwei Guo, and Hang Li. 2017b. Cascaded attention based unsupervised information distillation for compressive summarization. In *EMNLP*, pages 2081–2090.

Piji Li, Zihao Wang, Wai Lam, Zhaochun Ren, and Lidong Bing. 2017c. Salience estimation via variational auto-encoders for multi-document summarization. In *AAAI*, pages 3497–3503.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proc. of the ACL-04 workshop*, volume 8. Barcelona, Spain.

He Liu, Hongliang Yu, and Zhi-Hong Deng. 2015. Multi-document summarization based on two-level sparse representation model. In *AAAI*, pages 196–202.

Ulrike von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416.

Shulei Ma, Zhi-Hong Deng, and Yunlun Yang. 2016. An unsupervised multi-document summarization framework based on neural document model. In *COLING*, pages 1514–1523.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *EMNLP*, pages 404–411.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

Ani Nenkova, Lucy Vanderwende, and Kathleen R. McKeown. 2006. A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In *SIGIR*, pages 573–580.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *ICML*, pages 1310–1318.

Dragomir R Radev, Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: Sentence extraction, utility-based evaluation, and user studies. In *NAACL-ANLP, Vol. 4*, pages 21–30.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma, and Maarten de Rijke. 2017. Leveraging contextual sentence relations for extractive summarization using a neural attention model. In *SIGIR*, pages 95–104.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *ACL, Vol. 1*, pages 1073–1083.

Hongya Song, Zhaochun Ren, Shangsong Liang, Piji Li, Jun Ma, and Maarten de Rijke. 2017. Summarizing answers in non-factoid community question-answering. In *WSDM*, pages 405–414.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112.

Xiaojun Wan and Jianwu Yang. 2008. Multi-document summarization using cluster-based link analysis. In *SIGIR*, pages 299–306.

Kexiang Wang, Tianyu Liu, Zhifang Sui, and Baobao Chang. 2017. Affinity-preserving random walk for multi-document summarization. In *EMNLP*, pages 210–220.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir R. Radev. 2017. Graph-based neural multi-document summarization. In *CoNLL*, pages 452–462.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. In *ACL*, pages 654–663.