Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization

Sangwoo Cho, Logan Lebanoff, Hassan Foroosh, Fei Liu
Computer Science Department
University of Central Florida, Orlando, FL 32816, USA
{swcho,loganlebanoff}@knight.ucf.edu, {foroosh,feiliu}@cs.ucf.edu

Abstract

The most important obstacles facing multi-document summarization include excessive redundancy in source descriptions and the looming shortage of training data. These obstacles prevent encoder-decoder models from being used directly, but optimization-based methods such as determinantal point processes (DPPs) are known to handle them well. In this paper we seek to strengthen a DPP-based method for extractive multi-document summarization by presenting a novel similarity measure inspired by capsule networks. The approach measures redundancy between a pair of sentences based on surface form and semantic information. We show that our DPP system with improved similarity measure performs competitively, outperforming strong summarization baselines on benchmark datasets. Our findings are particularly meaningful for summarizing documents created by multiple authors containing redundant yet lexically diverse expressions.[1]

1 Introduction

Multi-document summarization is arguably one of the most important tools for information aggregation. It seeks to produce a succinct summary from a collection of textual documents created by multiple authors concerning a single topic (Nenkova and McKeown, 2011). The summarization technique has seen growing interest in a broad spectrum of domains that include summarizing product reviews (Gerani et al., 2014; Yang et al., 2018), student survey responses (Luo and Litman, 2015; Luo et al., 2016), forum discussion threads (Ding and Jiang, 2015; Tarnpradab et al., 2017), and news articles about a particular event (Hong et al., 2014). Despite the empirical success, most of the datasets remain small, and the cost of hiring human annotators to create ground-truth summaries for multi-document inputs can be prohibitive.

Impressive progress has been made on neural abstractive summarization using encoder-decoder models (Rush et al., 2015; See et al., 2017; Paulus et al., 2017; Chen and Bansal, 2018). These models, nonetheless, are data-hungry and learn poorly from small datasets, as is often the case with multi-document summarization. To date, studies have primarily focused on single-document summarization (See et al., 2017; Celikyilmaz et al., 2018; Kryscinski et al., 2018) and sentence summarization (Nallapati et al., 2016; Zhou et al., 2017; Cao et al., 2018; Song et al., 2018) in part because parallel training data are abundant and they can be conveniently acquired from the Web. Further, a notable issue with abstractive summarization is the reliability. These models are equipped with the capability of generating new words not present in the source. With greater freedom of lexical choices, the system summaries can contain inaccurate factual details and falsified content that prevent them from staying "true-to-original."

In this paper we instead focus on an extractive method exploiting the determinantal point process (DPP; Kulesza and Taskar, 2012) for multi-document summarization. DPP can be trained on small data, and because extractive summaries are free from manipulation, they largely remain true to the original. DPP selects a set of most representative sentences from the given source documents to form a summary, while maintaining high diversity among summary sentences. It is one of a family of optimization-based summarization methods that performed strongest in previous summarization competitions (Gillick and Favre, 2009; Lin and Bilmes, 2010; Kulesza and Taskar, 2011).

Diversity is an integral part of the DPP model. It is modelled by *pairwise repulsion* between sentences. In this paper we exploit the capsule net-

[1] Our code and data are publicly available at https://github.com/ucfnlp/summarization-dpp-capsnet

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1027–1038
Florence, Italy, July 28 - August 2, 2019.

works (Hinton et al., 2018) to measure pairwise sentence (dis)similarity, then leverage DPP to obtain a set of diverse summary sentences. Traditionally, the DPP method computes similarity scores based on the bag-of-words representation of sentences (Kulesza and Taskar, 2011) and with kernel methods (Gong et al., 2014). These methods, however, are incapable of capturing lexical and syntactic variations in the sentences (e.g., paraphrases), which are ubiquitous in multi-document summarization data as the source documents are created by multiple authors with distinct writing styles. We hypothesize that the recently proposed capsule networks, which learn high-level representations based on the orientational and spatial relationships of low-level components, can be a suitable supplement to model pairwise sentence similarity.

Importantly, we argue that predicting sentence similarity within the context of summarization has its uniqueness. It estimates if two sentences contain redundant information based on both surface word form and their underlying semantics. E.g., the two sentences "*Snowstorm slams eastern US on Friday*" and "*A strong wintry storm was dumping snow in eastern US after creating traffic havoc that claimed at least eight lives*" are considered similar because they carry redundant information and cannot both be included in the summary. These sentences are by no means semantically equivalent, nor do they exhibit a clear entailment relationship. The task thus should be distinguished from similar tasks such as predicting natural language inference (Bowman et al., 2015; Williams et al., 2018) or semantic textual similarity (Cer et al., 2017). In this work, we describe a novel method to collect a large amount of sentence pairs that are deemed similar for summarization purpose. We contrast this new dataset with those used for textual entailment for modeling sentence similarity and demonstrate its effectiveness on discriminating sentences and generating diverse summaries. The contributions of this work can be summarized as follows:

- we present a novel method inspired by the determinantal point process for multi-document summarization. The method includes a *diversity* measure assessing the redundancy between sentences, and a *quality* measure that indicates the importance of sentences. DPP extracts a set of summary sentences that are both representative of the document set and remain diverse;
- we present the first study exploiting capsule networks for determining sentence similarity for summarization purpose. It is important to recognize that summarization places particular emphasis on measuring redundancy between sentences; and this notion of similarity is different from that of entailment and semantic textual similarity (STS);
- our findings suggest that effectively modeling pairwise sentence similarity is crucial for increasing summary diversity and boosting summarization performance. Our DPP system with improved similarity measure performs competitively, outperforming strong summarization baselines on benchmark datasets.

2 Related Work

Extractive summarization approaches are the most popular in real-world applications (Carbonell and Goldstein, 1998; Daumé III and Marcu, 2006; Galanis and Androutsopoulos, 2010; Hong et al., 2014; Yogatama et al., 2015). These approaches focus on identifying representative sentences from a single document or set of documents to form a summary. The summary sentences can be optionally compressed to remove unimportant constituents such as prepositional phrases to yield a succinct summary (Knight and Marcu, 2002; Zajic et al., 2007; Martins and Smith, 2009; Berg-Kirkpatrick et al., 2011; Thadani and McKeown, 2013; Wang et al., 2013; Li et al., 2013, 2014; Filippova et al., 2015; Durrett et al., 2016). Extractive summarization methods are mostly unsupervised or lightly-supervised using thousands of training examples. Given its practical importance, we explore an extractive method in this work for multi-document summarization.

It is not uncommon to cast summarization as a discrete optimization problem (Gillick and Favre, 2009; Takamura and Okumura, 2009; Lin and Bilmes, 2010; Hirao et al., 2013). In this formulation, a set of binary variables are used to indicate whether their corresponding source sentences are to be included in the summary. The summary sentences are selected to maximize the coverage of important source content, while minimizing the summary redundancy and subject to a length constraint. The optimization can be performed using an off-the-shelf tool such as Gurobi, IBM CPLEX, or via a greedy approximation algorithm. Notable optimization frameworks include integer linear

programming (Gillick and Favre, 2009), determinantal point processes (Kulesza and Taskar, 2012), submodular functions (Lin and Bilmes, 2010), and minimum dominating set (Shen and Li, 2010). In this paper we employ the DPP framework because of its remarkable performance on various summarization problems (Zhang et al., 2016).

Recent years have also seen considerable interest in neural approaches to summarization. In particular, neural extractive approaches focus on learning vector representations of source sentences; then based on these representations they determine if a source sentence is to be included in the summary (Cheng and Lapata, 2016; Yasunaga et al., 2017; Nallapati et al., 2017; Narayan et al., 2018). Neural abstractive approaches usually include an encoder used to convert the entire source document to a continuous vector, and a decoder for generating an abstract word by word conditioned on the document vector (Paulus et al., 2017; Tan et al., 2017; Guo et al., 2018; Kedzie et al., 2018). These neural models, however, require large training data containing hundreds of thousands to millions of examples, which are still unavailable for the multi-document summarization task. To date, most neural summarization studies are performed for single document summarization.

Extracting summary-worthy sentences from the source documents is important even if the ultimate goal is to generate abstracts. Recent abstractive studies recognize the importance of separating "salience estimation" from "text generation" so as to reduce the amount of training data required by encoder-decoder models (Gehrmann et al., 2018; Lebanoff et al., 2018, 2019). An extractive method is often leveraged to identify salient source sentences, then a neural text generator rewrites the selected sentences into an abstract. Our pursuit of the DPP method is especially meaningful in this context. As described in the next section, DPP has an extraordinary ability to distinguish redundant descriptions, thereby avoiding passing redundant content to the abstractor that can cause an encoder-decoder model to fail.

3 The DPP Framework

Let $\mathcal{Y} = \{1, 2, \cdots, \mathsf{N}\}$ be a ground set containing N items, corresponding to all sentences of the source documents. Our goal is to identify a subset of items $Y \subseteq \mathcal{Y}$ that forms an extractive summary of the document set. A determinantal point process (DPP; Kulesza and Taskar, 2012) defines a probability measure over all subsets of $\mathcal{Y}$ s.t.

$$\mathcal{P}(Y; L) = \frac{\det(L_Y)}{\det(L + I)}, \tag{1}$$

$$\sum_{Y \subseteq \mathcal{Y}} \det(L_Y) = \det(L + I), \tag{2}$$

where $\det(\cdot)$ is the determinant of a matrix; I is the identity matrix; $L \in \mathbb{R}^{\mathsf{N} \times \mathsf{N}}$ is a positive semidefinite matrix, known as the L-ensemble; L_{ij} measures the correlation between sentences i and j; and L_Y is a submatrix of L containing only entries indexed by elements of Y. Finally, the probability of an extractive summary $Y \subseteq \mathcal{Y}$ is proportional to the determinant of the matrix L_Y (Eq. (1)).

Kulesza and Taskar (2012) provide a decomposition of the L-ensemble matrix: $L_{ij} = q_i \cdot S_{ij} \cdot q_j$ where $q_i \in \mathbb{R}^+$ is a positive real number indicating the *quality* of a sentence; and S_{ij} is a measure of *similarity* between sentences i and j. This formulation separately models the sentence quality and pairwise similarity before combining them into a unified model. Let $Y = \{i, j\}$ be a summary containing only two sentences i and j, its probability $\mathcal{P}(Y; L)$ can be computed as

$$\begin{aligned} \mathcal{P}(Y = \{i, j\}; L) &\propto \det(L_Y) \\ &= \begin{vmatrix} q_i S_{ii} q_i & q_i S_{ij} q_j \\ q_j S_{ji} q_i & q_j S_{jj} q_j \end{vmatrix} \\ &= q_i^2 \cdot q_j^2 \cdot (1 - S_{ij}^2). \end{aligned} \tag{3}$$

Eq. (3) indicates that, if sentence i is of high quality, denoted by q_i, then any summary containing it will have high probability. If two sentences i and j are similar to each other, denoted by S_{ij}, then any summary containing both sentences will have low probability. The summary Y achieving the highest probability thus should contain a set of high-quality sentences while maintaining high diversity among the selected sentences (via pairwise repulsion). $\det(L_Y)$ also has a particular geometric interpretation as the squared volume of the space spanned by sentence vectors i and j, where the quality measure indicates the length of the vector and the similarity indicates the angle between two vectors (Figure 1).

We adopt a feature-based approach to compute sentence quality: $q_i = \exp(\boldsymbol{\theta}^\top \mathbf{x}_i)$. In particular, $\mathbf{x}_i$ is a feature vector for sentence i and $\boldsymbol{\theta}$ are the feature weights to be learned during training. Kulesza and Taskar (2011) define sentence similarity as $S_{i,j} = \boldsymbol{\phi}_i^\top \boldsymbol{\phi}_j$, where $\|\boldsymbol{\phi}_i\|_2 = 1$ $(\forall i)$ is

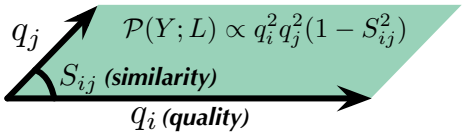

Figure 1: The DPP model specifies the probability of a summary $\mathcal{P}(Y = \{i, j\}; L)$ to be proportional to the squared volume of the space spanned by sentence vectors i and j.

a sentence TF-IDF vector. The model parameters $\boldsymbol{\theta}$ are optimized by maximizing the log-likelihood of training data (Eq. (4)) and this objective can be optimized efficiently with subgradient descent.[2]

$$\boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}} \sum_{m=1}^{M} \log \mathcal{P}(\hat{Y}^{(m)}; L(\mathcal{Y}^{(m)}; \boldsymbol{\theta})) \quad (4)$$

During training, we create the ground-truth extractive summary ($\hat{Y}$) for a document set based on human reference summaries (abstracts) using the following procedure. At each iteration we select a source sentence sharing the longest common subsequence with the human reference summaries; the shared words are then removed from human summaries to avoid duplicates in future selection. Similar methods are exploited by Nallapati et al. (2017) and Narayan et al. (2018) to create ground-truth extractive summaries. At test time, we perform inference using the learned DPP model to obtain a system summary (Y). We implement a greedy method (Kulesza and Taskar, 2012) to iteratively add a sentence to the summary so that $\mathcal{P}(Y; L)$ yields the highest probability (Eq. (1)), until a summary length limit is reached.

For the DPP framework to be successful, the sentence similarity measure (S_{ij}) has to accurately capture if any two sentences contain redundant information. This is especially important for multi-document summarization as redundancy is ubiquitous in source documents. The source descriptions frequently contain redundant yet lexically diverse expressions such as sentential paraphrases where people write about the same event using distinct styles (Hu et al., 2019). Without accurately modelling sentence similarity, redundant content can make their way into the summary and further prevent useful information from being included given the summary length limit. Existing cosine similarity measure between sentence TF-IDF vectors can be incompetent in modeling semantic relatedness. In the following section we exploit the recently introduced capsule networks (Hinton et al., 2018) to measure pairwise sentence similarity; it considers if two sentences share any words in common and more importantly the semantic closeness of sentence descriptions.

[2]The sentence features include the length and position of a sentence, the cosine similarity between sentence and document TF-IDF vectors (Kulesza and Taskar, 2011). We refrain from using sophisticated features to avoid model overfitting.

4 An Improved Similarity Measure

Our goal is to develop an advanced similarity measure for pairs of sentences such that semantically similar sentences can receive high scores despite that they have very few words in common. E.g., "*Snowstorm slams eastern US on Friday*" and "*A strong wintry storm was dumping snow in eastern US after creating traffic havoc that claimed at least eight lives*" have only two words in common. Nonetheless, they contain redundant information and cannot both be included in the summary.

Let $\{\mathbf{x}^{\mathsf{a}}, \mathbf{x}^{\mathsf{b}}\} \in \mathbb{R}^{\mathsf{E} \times \mathsf{L}}$ denote two sentences a and b. Each consists of a sequence of word embeddings, where E is the embedding size and L is the sentence length with zero-padding to the right for shorter sentences. A convolutional layer with multiple filter sizes is first applied to each sentence to extract local features (Eq. (5)), where $\mathbf{x}^{\mathsf{a}}_{i:i+k-1} \in \mathbb{R}^{k\mathsf{E}}$ denotes a flattened embedding for position i with a filter size k, and $\mathbf{u}^{\mathsf{a}}_{i,k} \in \mathbb{R}^d$ is the resulting local feature for position i; f is a non-linear activation function (e.g., ReLU); $\{\mathbf{W}^u, \mathbf{b}^u\}$ are model parameters.

$$\mathbf{u}^{\mathsf{a}}_{i,k} = f(\mathbf{W}^u \mathbf{x}^{\mathsf{a}}_{i:i+k-1} + \mathbf{b}^u) \quad (5)$$

We use $\mathbf{u}^{\mathsf{a}}_i \in \mathbb{R}^{\mathsf{D}}$ to denote the concatenation of local features generated using various filter sizes. Following Kim et al. (2014), we employ filter sizes $k \in \{3, 4, 5, 6, 7\}$ with an equal number of filters (d) for each size ($\mathsf{D} = 5d$). After applying max-pooling to local features of all positions, we obtain a representation $\mathbf{u}^{\mathsf{a}} = \text{max-pooling}(\mathbf{u}^{\mathsf{a}}_i) \in \mathbb{R}^{\mathsf{D}}$ for sentence a; and similarly we obtain $\mathbf{u}^{\mathsf{b}} \in \mathbb{R}^{\mathsf{D}}$ for sentence b. It is not uncommon for state-of-the-art sentence similarity classifiers (Chen et al., 2018) to concatenate the two sentence vectors, their absolute difference and element-wise product $[\mathbf{u}^{\mathsf{a}}; \mathbf{u}^{\mathsf{b}}; |\mathbf{u}^{\mathsf{a}} - \mathbf{u}^{\mathsf{b}}|; \mathbf{u}^{\mathsf{a}} \circ \mathbf{u}^{\mathsf{b}}]$, and feed this representation to a fully connected layer to predict if two sentences are similar.

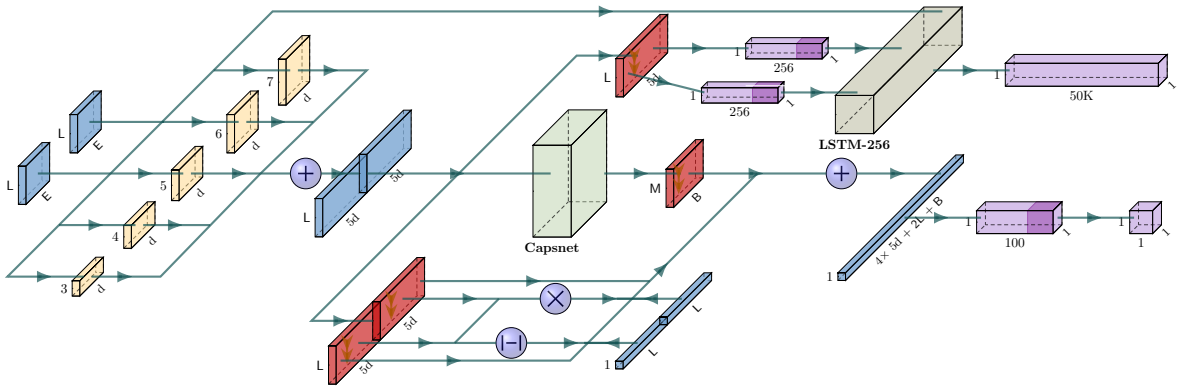

Figure 2: The system architecture utilizing CapsNet for predicting sentence similarity. denotes the inputs and intermediate outputs; the convolutional layer; max-pooling layer; fully-connected layer; and ReLU activation.

Nevertheless, we conjecture that such representation may be insufficient to fully characterize the relationship between components of the sentences in order to model sentence similarity. For example, the term "*snowstorm*" in sentence a is semantically related to "*wintry storm*" and "*dumping snow*" in sentence b; this low-level interaction indicates that the two sentences contain redundant information and it cannot be captured by the above model. Importantly, the capsule networks proposed by Hinton et al. (2018) are designed to characterize the spatial and orientational relationships between low-level components. We thus seek to exploit CapsNet to strengthen the capability of our system for identifying redundant sentences.

Let $\{\mathbf{u}_i^{\mathsf{a}}, \mathbf{u}_i^{\mathsf{b}}\}_{i=1}^{\mathsf{L}} \in \mathbb{R}^{\mathsf{D}}$ be low-level representations (i.e., capsules). We seek to transform them to high-level capsules $\{\mathbf{v}_j\}_{j=1}^{\mathsf{M}} \in \mathbb{R}^{\mathsf{B}}$ that characterize the interaction between low-level components. Each low-level capsule $\mathbf{u}_i \in \mathbb{R}^{\mathsf{D}}$ is multiplied by a linear transformation matrix to dedicate a portion of it, denoted by $\hat{\mathbf{u}}_{j|i} \in \mathbb{R}^{\mathsf{B}}$, to the construction of a high-level capsule j (Eq. (6)); where $\{\mathbf{W}_{ij}^v\} \in \mathbb{R}^{\mathsf{D}\times\mathsf{B}}$ are model parameters. To reduce parameters and prevent overfitting, we further encourage sharing parameters over all low-level capsules, yielding $\mathbf{W}_{1j}^v = \mathbf{W}_{2j}^v = \cdots$, and the same parameter sharing is described in (Zhao et al., 2018). By computing the weighted sum of $\hat{\mathbf{u}}_{j|i}$, whose weights c_{ij} indicate the strength of interaction between a low-level capsule i and a high-level capsule j, we obtain an (unnormalized) capsule (Eq. (7)); we then apply a nonlinear squash function $g(\cdot)$ to normalize the length the vector to be less than 1, yielding $\mathbf{v}_j \in \mathbb{R}^{\mathsf{B}}$.

$$\hat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij}^v \mathbf{u}_i \tag{6}$$

$$\mathbf{v}_j = g\Big(\sum_i c_{ij}\hat{\mathbf{u}}_{j|i}\Big) \tag{7}$$

Routing (Sabour et al., 2017; Zhao et al., 2019) aims to adjust the interaction weights (c_{ij}) using an iterative, EM-like method. Initially, we set $\{b_{ij}\}$ to be zero for all i and j. Per Eq. (8), $\mathbf{c}_i$ becomes a uniform distribution indicating a low-level capsule i contributes equally to all its upper level capsules. After computing $\hat{\mathbf{u}}_{j|i}$ and $\mathbf{v}_j$ using Eq. (6-7), the weights b_{ij} are updated according to the strength of interaction (Eq. (9)). If $\hat{\mathbf{u}}_{j|i}$ agrees with a capsule $\mathbf{v}_j$, their interaction weight will be increased, and decreased otherwise. This process is repeated for r iterations to stabilize c_{ij}.

$$\mathbf{c}_i \leftarrow \text{softmax}(\mathbf{b}_i) \tag{8}$$

$$b_{ij} \leftarrow b_{ij} + \hat{\mathbf{u}}_{j|i}\mathbf{v}_j \tag{9}$$

The high-level capsules $\{\mathbf{v}_j\}_{j=1}^{\mathsf{M}}$ effectively encode spatial and orientational relationships of low-level capsules. To identify the most prominent interactions, we apply max-pooling to all high-level capsules to produce $\mathbf{v} = \text{max-pooling}_j(\mathbf{v}_j) \in \mathbb{R}^{\mathsf{B}}$. This representation $\mathbf{v}$, aimed to encode interactions between sentences a and b, is concatenated with $[\mathbf{u}^{\mathsf{a}}; \mathbf{u}^{\mathsf{b}}; |\mathbf{u}^{\mathsf{a}} - \mathbf{u}^{\mathsf{b}}|; \mathbf{u}^{\mathsf{a}} \circ \mathbf{u}^{\mathsf{b}}]$ and binary vectors $[\mathbf{z}^{\mathsf{a}}; \mathbf{z}^{\mathsf{b}}]$ that indicate if any word in sentence a appears in sentence b and vice versa; they are used as input to a fully connected layer to predict if a pair of sentences contain redundant information. Our loss function contains two components, including a binary cross-entropy loss indicating whether the

prediction is correct or not, and a reconstruction loss for reconstructing a sentence a conditioned on $\mathbf{u}^{\mathsf{a}}$ by predicting one word at a time using a recurrent neural network, and similarly for sentence b. A hyperparameter λ is used to balance contributions from both sides. In Figure 2 we present an overview of the system architecture, and hyperparameters are described in the supplementary.

5 Datasets

To our best knowledge, there is no dataset focusing on determining if two sentences contain redundant information. It is a nontrivial task in the context of multi-document summarization. Further, we argue that the task should be distinguished from other semantic similarity tasks: semantic textual similarity (STS; Cer et al., 2017) assesses to what degree two sentences are semantically equivalent to each other; natural language inference (NLI; Bowman et al., 2015) determines if one sentence ("hypothesis") can be semantically inferred from the other sentence ("premise"). Nonetheless, redundant sentences found in a set of source documents discussing a particular topic are not necessarily semantically equivalent or express an entailment relationship. We compare different datasets in §6.

Sentence redundancy dataset A novel dataset containing over 2 million sentence pairs is introduced in this paper for sentence redundancy prediction. We hypothesize that it is likely for a summary sentence and its most similar source sentence to contain redundant information. Because humans create summaries using generalization, paraphrasing, and other high-level text operations, a summary sentence and its source sentence can be semantically similar, yet contain diverse expressions. Fortunately, such source/summary sentence pairs can be conveniently derived from single-document summarization data. We analyze the CNN/Daily Mail dataset (Hermann et al., 2015) that contains a massive collection of single news articles and their human-written summaries. For each summary sentence, we identify its most similar source sentence by calculating the averaged R-1, R-2, and R-L F-scores (Lin, 2004) between a source and summary sentences. We consider a summary sentence to have no match if the score is lower than a threshold. We obtain negative examples by randomly sampling two sentences from a news article. In total, our training / dev / test sets contain 2,084,798 / 105,936 / 86,144 sentence pairs and we make the dataset available to advance research on sentence redundancy.

System	DUC-04 R-1	DUC-04 R-2	DUC-04 R-SU4
Opinosis (Ganesan et al., 2010)	27.07	5.03	8.63
Extract+Rewrite (Song et al., 2018)	28.90	5.33	8.76
Pointer-Gen (See et al., 2017)	31.43	6.03	10.01
SumBasic (Vanderwende et al., 2007)	29.48	4.25	8.64
KLSumm (Haghighi et al., 2009)	31.04	6.03	10.23
LexRank (Erkan and Radev, 2004)	34.44	7.11	11.19
Centroid (Hong et al., 2014)	35.49	7.80	12.02
ICSISumm (Gillick and Favre, 2009)	37.31	9.36	13.12
DPP (Kulesza and Taskar, 2011)†	38.10	9.14	13.40
DPP-Capsnet (this work)	38.25	9.22	13.40
DPP-Combined (this work)	**39.35**	**10.14**	**14.15**

Table 1: ROUGE results on DUC-04. † indicates our reimplementation of Kulesza and Taskar (2011).

Summarization datasets We evaluate our DPP-based system on benchmark multi-document summarization datasets. The task is to create a succinct summary with up to 100 words from a cluster of 10 news articles discussing a single topic. The DUC and TAC datasets (Over and Yen, 2004; Dang and Owczarzak, 2008) have been used in previous summarization competitions. In this paper we use DUC-03/04 and TAC-08/09/10/11 datasets that contain 60/50/48/44/46/44 document clusters respectively. Four human reference summaries have been created for each document cluster by NIST assessors. Any system summaries are evaluated against human reference summaries using the ROUGE software (Lin, 2004)[3], where R-1, -2, and -SU4 respectively measure the overlap of unigrams, bigrams, unigrams and skip bigrams with a maximum distance of 4 words. We report results on DUC-04 (trained on DUC-03) and TAC-11 (trained on TAC-08/09/10) that are often used as standard test sets (Hong et al., 2014).

6 Experimental Results

In this section we discuss results that we obtained for multi-document summarization and determining redundancy between sentences.

6.1 Summarization Results

We compare our system with a number of strong summarization baselines (Table 1 and 2). In particular, *SumBasic* (Vanderwende et al., 2007) is an extractive approach assuming words occurring fre-

[3] w/ options -n 2 -m -w 1.2 -c 95 -r 1000 -l 100

System	TAC-11 R-1	R-2	R-SU4
Opinosis (Ganesan et al., 2010)	25.15	5.12	8.12
Extract+Rewrite (Song et al., 2018)	29.07	6.11	9.20
Pointer-Gen (See et al., 2017)	31.44	6.40	10.20
SumBasic (Vanderwende et al., 2007)	31.58	6.06	10.06
KLSumm (Haghighi et al., 2009)	31.23	7.07	10.56
LexRank (Erkan and Radev, 2004)	33.10	7.50	11.13
DPP (Kulesza and Taskar, 2011)†	36.95	9.83	13.57
DPP-Capsnet (this work)	36.61	9.30	13.09
DPP-Combined (this work)	**37.30**	**10.13**	**13.78**

Table 2: ROUGE results on the TAC-11 dataset.

quently in a document cluster are more likely to be included in the summary; *KL-Sum* (Haghighi and Vanderwende, 2009) is a greedy approach adding a sentence to the summary to minimize KL divergence; and *LexRank* (Erkan and Radev, 2004) is a graph-based approach computing sentence importance based on eigenvector centrality.

We additionally consider abstractive baselines to illustrate how well these systems perform on multi-document summarization: *Opinosis* (Ganesan et al., 2010) focuses on creating a word co-occurrence graph from the source documents and searching for salient graph paths to create an abstract; *Extract+Rewrite* (Song et al., 2018) selects sentences using LexRank and condenses each sentence to a title-like summary; *Pointer-Gen* (See et al., 2017) seeks to generate abstracts by copying words from the source documents and generating novel words not present in the source text.

Our DPP-based framework belongs to a strand of optimization-based methods. In particular, *ICSISumm* (Gillick et al., 2009) formulates extractive summarization as integer linear programming; it identifies a globally-optimal set of sentences covering the most important concepts of the source documents; *DPP* (Kulesza and Taskar, 2011) selects an optimal set of sentences that are representative of the source documents and with maximum diversity, as determined by the determinantal point process. Gong et al. (2014) show that the DPP performs well on summarizing both text and video.

We experiment with several variants of the DPP model: *DPP-Capsnet* computes the similarity between sentences (S_{ij}) using the CapsNet described in Sec. §4 and trained using our newly-constructed sentence redundancy dataset, whereas the default DPP framework computes sentence similarity as the cosine similarity of sentence TF-IDF vectors. *DPP-Combined* linearly combines the cosine similarity with the CapsNet output using an interpolation coefficient determined on the dev set[4].

Table 1 and 2 illustrate the summarization results we have obtained for the DUC-04 and TAC-11 datasets. Our DPP methods perform superior to both extractive and abstractive baselines, indicating the effectiveness of optimization-based methods for extractive multi-document summarization. The DPP optimizes for summary sentence selection to maximize their content coverage and diversity, expressed as the squared volume of the space spanned by the selected sentences.

Further, we observe that the DPP system with combined similarity metrics yields the highest performance, achieving 10.14% and 10.13% F-scores respectively on DUC-04 and TAC-11. This finding suggests that the cosine similarity of sentence TF-IDF vectors and the CapsNet semantic similarity successfully complement each other to provide the best overall estimate of sentence redundancy. A close examination of the system outputs reveal that important topical words (e.g., "*$3 million*") that are frequently discussed in the document cluster can be crucial for determining sentence redundancy, because sentences sharing the same topical words are more likely to be considered redundant. While neural models such as the CapsNet rarely explicitly model word frequencies, the TF-IDF sentence representation is highly effective in capturing topical terms.

In Table 3 we show example system summaries and a human-written reference summary. We observe that LexRank tends to extract long and comprehensive sentences that yield high graph centrality; the abstractive pointer-generator networks, despite the promising results, can sometimes fail to generate meaningful summaries (e.g., "*a third of all 3-year-olds ⋯ have been given to a child*"). In contrast, our DPP method is able to select a balanced set of representative and diverse summary sentences. We next compare several semantic similarity datasets to gain a better understanding of modeling sentence redundancy for summarization.

6.2 Sentence Similarity

We compare three standard datasets used for semantic similarity tasks, including *SNLI* (Bowman et al., 2015), used for natural language inference, *STS-Benchmark* (Cer et al., 2017) for semantic

[4]The Capsnet coefficient λ_c is selected to be 0.2 and 0.1 respectively for the DUC-04 and TAC-11 dataset.

LexRank Summary

- The official, Dr. Charles J. Ganley, director of the office of nonprescription drug products at the Food and Drug Administration, said in an interview that the agency was "revisiting the risks and benefits of the use of these drugs in children" and that "we're particularly concerned about the use of these drugs in children less than 2 years of age."
- The Consumer Healthcare Products Association, an industry trade group that has consistently defended the safety of pediatric cough and cold medicines, recommended in its own 156-page safety review, also released Friday, that the FDA consider mandatory warning labels saying that they should not be used in children younger than two.
- Major makers of over-the-counter infant cough and cold medicines announced Thursday that they were voluntarily withdrawing their products from the market for fear that they could be misused by parents.

Pointer-Gen Summary

- Dr. Charles Ganley, a top food and drug administration official, said the agency was "revisiting the risks and benefits of the use of these drugs in children," the director of the FDA's office of nonprescription drug products.
- The FDA will formally consider revising labeling at a meeting scheduled for Oct. 18-19.
- The withdrawal comes two weeks after reviewing reports of side effects over the last four decades, a 1994 study found that more than a third of all 3-year-olds in the United States were estimated to have been given to a child.

DPP-Combined Summary

- Johnson & Johnson on Thursday voluntarily recalled certain infant cough and cold products, citing "rare" instances of misuse leading to overdoses.
- Federal drug regulators have started a broad review of the safety of popular cough and cold remedies meant for children, a top official said Thursday.
- Safety experts for the Food and Drug Administration urged the agency on Friday to consider an outright ban on over-the-counter, multi-symptom cough and cold medicines for children under 6.
- Major makers of over-the-counter infant cough and cold medicines announced Thursday that they were voluntarily withdrawing their products from the market for fear that they could be misused by parents.

Human Reference Summary

- On March 1, 2007, the Food/Drug Administration (FDA) started a broad safety review of children's cough/cold remedies.
- They are particularly concerned about use of these drugs by infants.
- By September 28th, the 356-page FDA review urged an outright ban on all such medicines for children under six.
- Dr. Charles Ganley, a top FDA official said "We have no data on these agents of what's a safe and effective dose in Children." The review also stated that between 1969 and 2006, 123 children died from taking decongestants and antihistimines.
- On October 11th, all such infant products were pulled from the markets.

Table 3: Example system summaries and the human reference summary. LexRank extracts long and comprehensive sentences that yield high graph centrality. Pointer-Gen (abstractive) has difficulty in generating faithful summaries (see the last bullet "*all 3-year-olds ... have been given to a child*"). DPP is able to select a balanced set of representative and diverse sentences.

Dataset	Train	Dev	Test	Accu.
STS-Benchmark (Cer et al., 2017)	5,749	1,500	1,379	64.7%
SNLI (Bowman et al., 2015)	366,603	6,607	6,605	93.0%
Src-Summ Pairs (this work)	2,084,798	105,936	86,144	**94.8**%

Table 4: Sentence similarity datasets and CapsNet's performance on them. SNLI discriminates between entailment and contradiction; STS is pretrained using Src-Summ pairs and fine-tuned on its train split.

STS-Benchmark (a) *Four girls happily walk down a sidewalk.* **(b)** *Three young girls walk down a sidewalk.* ✗

SNLI (a) *3 young man in hoods standing in the middle of a quiet street facing the camera.* **(b)** *Three hood wearing people pose for a picture.* ✓

Src-Summ Pairs (a) *He ended up killing five girls and wounding five others before killing himself.* **(b)** *Nearly four months ago, a milk delivery-truck driver lined up 10 girls in a one-room schoolhouse in this Amish farming community and opened fire, killing five of them and wounding five others before turning the gun on himself.* ✓

Table 5: Example positive (✓) and negative (✗) sentence pairs from the semantic similarity datasets.

equivalence, and our newly-constructed *Src-Summ* sentence pairs. Details are presented in Table 4.

We observe that CapsNet achieves the highest prediction accuracy of 94.8% on the *Src-Summ* dataset and it yields similar performance on *SNLI*, indicating the effectiveness of CapsNet on characterizing semantic similarity. *STS* appears to be a more challenging task, where CapsNet yields 64.7% accuracy. Note that we perform two-way classification on SNLI to discriminate entailment and contradiction. The STS dataset is too small to be used to train CapsNet without overfitting, we thus pre-train the model on *Src-Summ* pairs, and use the train split of *STS* to fine-tune parameters.

Table 5 shows example positive and negative sentence pairs from the *STS*, *SNLI*, and *Src-Summ* datasets. The *STS* and *SNLI* datasets are constructed by human annotators to test a system's capability of learning sentence representations. The sentences can share very few words in common but still express an entailment relationship (positive); or the sentences can share a lot of words in common yet they are semantically distinct (negative). These cases are usually not seen in summarization datasets containing clusters of documents discussing single topics. The *Src-Summ* dataset successfully strike a balance between shar-

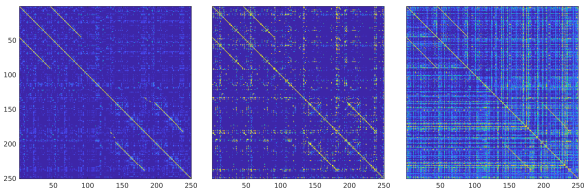

Figure 3: Heatmaps for topic D31008 of DUC-04 (cropped to 200 sentences) that shows the cosine similarity score of sentence TF-IDF vectors (*Cosine*, left), and the CapsNet output trained respectively on *SNLI* (right) and *Src-Summ* (middle) datasets. The short off-diagonal lines are near-identical sentences found in the document cluster.

ing common words yet containing diverse expressions. It is thus a good fit for training classifiers to detect sentence redundancy.

Figure 3 compares heatmaps generated by computing cosine similarity of sentence TF-IDF vectors (*Cosine*), and training CapsNet on *SNLI* and *Src-Summ* datasets respectively. We find that the *Cosine* similarity scores are relatively strict, as a vast majority of sentence pairs are assigned zero similarity, because these sentences have no word overlap. At the other extreme, *CapsNet+SNLI* labels a large quantity of sentence pairs as false positives, because its training data frequently contain sentences that share few words in common but nonetheless are positive, i.e., expressing an entailment relationship. The similarity scores generated by *CapsNet+SrcSumm* are more moderate comparing to *CapsNet+SNLI* and *Cosine*, suggesting the appropriateness of using *Src-Summ* sentence pairs for estimating sentence redundancy.

7 Conclusion

We strengthen a DPP-based multi-document summarization system with improved similarity measure inspired by capsule networks for determining sentence redundancy. We show that redundant sentences not only have common words but they can be semantically similar with little word overlap. Both aspects should be modelled in calculating pairwise sentence similarity. Our system yields competitive results on benchmark datasets surpassing strong summarization baselines.

Acknowledgments

The authors are grateful to the reviewers for their insightful feedback. We would also like to extend our thanks to Boqing Gong, Xiaodan Zhu and Fei Sha for useful discussions.

References

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. Jointly learning to extract and compress. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, , and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and cross-lingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval)*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. Neural natural language inference models enhanced with external knowledge. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of ACL*.

Hoa Trang Dang and Karolina Owczarzak. 2008. Overview of the TAC 2008 update summarization task. In *Proceedings of Text Analysis Conference (TAC)*.

Hal Daumé III and Daniel Marcu. 2006. Bayesian query-focused summarization. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Ying Ding and Jing Jiang. 2015. Towards opinion summarization from online forums. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP)*.

Greg Durrett, Taylor Berg-Kirkpatrick, and Dan Klein. 2016. Learning-based single-document summarization with compression and anaphoricity constraints. In *Proceedings of the Association for Computational Linguistics (ACL)*.

Günes Erkan and Dragomir R. Radev. 2004. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*.

Katja Filippova, Enrique Alfonseca, Carlos Colmenares, Lukasz Kaiser, and Oriol Vinyals. 2015. Sentence compression by deletion with lstms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Dimitrios Galanis and Ion Androutsopoulos. 2010. An extractive supervised two-stage method for sentence compression. In *Proceedings of NAACL-HLT*.

Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. 2010. Opinosis: A graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the International Conference on Computational Linguistics (COLING)*.

Sebastian Gehrmann, Yuntian Deng, and Alexander M. Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Shima Gerani, Yashar Mehdad, Giuseppe Carenini, Raymond T. Ng, and Bita Nejat. 2014. Abstractive summarization of product reviews using discourse structure. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *Proceedings of the NAACL Workshop on Integer Linear Programming for Natural Langauge Processing*.

Boqing Gong, Wei-Lun Chao, Kristen Grauman, and Fei Sha. 2014. Diverse sequential subset selection for supervised video summarization. In *Proceedings of Neural Information Processing Systems (NIPS)*.

Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018. Soft, layer-specific multi-task summarization with entailment and question generation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of Neural Information Processing Systems (NIPS)*.

Geoffrey Hinton, Sara Sabour, and Nicholas Frosst. 2018. Matrix capsules with EM routing. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Tsutomu Hirao, Yasuhisa Yoshida, Masaaki Nishino, Norihito Yasuda, and Masaaki Nagata. 2013. Single-document summarization as a tree knapsack problem. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Kai Hong, John M Conroy, Benoit 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 *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*.

J. Edward Hu, Rachel Rudinger, Matt Post, and Benjamin Van Durme. 2019. ParaBank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*.

Chris Kedzie, Kathleen McKeown, and Hal Daume III. 2018. Content selection in deep learning models of summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Gunhee Kim, Leonid Sigal, and Eric P. Xing. 2014. Joint summarization of large-scale collections of web images and videos for storyline reconstruction. In *Proceedings of CVPR*.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*.

Wojciech Kryscinski, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Alex Kulesza and Ben Taskar. 2011. Learning determinantal point processes. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*.

Alex Kulesza and Ben Taskar. 2012. *Determinantal Point Processes for Machine Learning*. Now Publishers Inc.

Logan Lebanoff, Kaiqiang Song, Franck Dernoncourt, Doo Soon Kim, Seokhwan Kim, Walter Chang, and Fei Liu. 2019. Scoring sentence singletons and pairs

for abstractive summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Logan Lebanoff, Kaiqiang Song, and Fei Liu. 2018. Adapting the neural encoder-decoder framework from single to multi-document summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Chen Li, Fei Liu, Fuliang Weng, and Yang Liu. 2013. Document summarization via guided sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Chen Li, Yang Liu, Fei Liu, Lin Zhao, and Fuliang Weng. 2014. Improving multi-document summarization by sentence compression based on expanded constituent parse tree. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*.

Chin-Yew Lin. 2004. ROUGE: a package for automatic evaluation of summaries. In *Proceedings of ACL Workshop on Text Summarization Branches Out*.

Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Proceedings of NAACL*.

Wencan Luo and Diane Litman. 2015. Summarizing student responses to reflection prompts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Wencan Luo, Fei Liu, Zitao Liu, and Diane Litman. 2016. Automatic summarization of student course feedback. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Andre F. T. Martins and Noah A. Smith. 2009. Summarization with a joint model for sentence extraction and compression. In *Proceedings of the ACL Workshop on Integer Linear Programming for Natural Language Processing*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of SIGNLL*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Ani Nenkova and Kathleen McKeown. 2011. Automatic summarization. *Foundations and Trends in Information Retrieval*.

Paul Over and James Yen. 2004. An introduction to DUC-2004. *National Institute of Standards and Technology*.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for sentence summarization. In *Proceedings of EMNLP*.

Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. 2017. Dynamic routing between capsules. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS)*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Chao Shen and Tao Li. 2010. Multi-document summarization via the minimum dominating set. In *Proceedings of the International Conference on Computational Linguistics (COLING)*.

Kaiqiang Song, Lin Zhao, and Fei Liu. 2018. Structure-infused copy mechanisms for abstractive summarization. In *Proceedings of the International Conference on Computational Linguistics (COLING)*.

Hiroya Takamura and Manabu Okumura. 2009. Text summarization model based on maximum coverage problem and its variant. In *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Sansiri Tarnpradab, Fei Liu, and Kien A. Hua. 2017. Toward extractive summarization of online forum discussions via hierarchical attention networks. In *Proceedings of the 30th Florida Artificial Intelligence Research Society Conference (FLAIRS)*.

Kapil Thadani and Kathleen McKeown. 2013. Sentence compression with joint structural inference. In *Proceedings of CoNLL*.

Lucy Vanderwende, Hisami Suzuki, Chris Brockett, and Ani Nenkova. 2007. Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing and Management*, 43(6):1606–1618.

Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. 2013. A sentence compression based framework to query-focused multi-document summarization. In *Proceedings of ACL*.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Min Yang, Qiang Qu, Ying Shen, Qiao Liu, Wei Zhao, and Jia Zhu. 2018. Aspect and sentiment aware abstractive review summarization. *Proceedings of the International Conference on Computational Linguistics (COLING)*.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*.

Dani Yogatama, Fei Liu, and Noah A. Smith. 2015. Extractive summarization by maximizing semantic volume. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*.

David Zajic, Bonnie J. Dorr, Jimmy Lin, and Richard Schwartz. 2007. Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. *Information Processing and Management*.

Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. 2016. Video summarization with long short-term memory. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Wei Zhao, Haiyun Peng, Steffen Eger, Erik Cambria, and Min Yang. 2019. Towards scalable and generalizable capsule network and its NLP applications. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Wei Zhao, Jianbo Ye, Min Yang, Zeyang Lei, Soufei Zhang, and Zhou Zhao. 2018. Investigating capsule networks with dynamic routing for text classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Qingyu Zhou, Nan Yang, Furu Wei, and Ming Zhou. 2017. Selective encoding for abstractive sentence summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

A Supplemental Material

In this section we summarize the hyperparameters used for the capsule networks. They include: the embedding size E is set to 300 dimensions; the maximum sentence length L is 44 words; in the convolutional layer we use d=100 filters for each filter size, and there are 5 filter sizes in total: $k \in \{3, 4, 5, 6, 7\}$. The number of high-level capsules M is set to 12, and the dimension of capsules B is set to 30, both are tuned on the development set. The dynamic routing process is repeated for r=3 iterations, following (Sabour et al., 2017). Further, the coefficient λ for the reconstruction loss term is set to 5e-5. We use a vocabulary of 50K words for reconstructing the sentences; they are the most frequently appearing words of the dataset.