Multi-Document Summarization with Determinantal Point Processes and Contextualized Representations

Sangwoo Cho♠, Chen Li♢, Dong Yu♢, Hassan Foroosh♠, Fei Liu♠

♠Computer Science Department, University of Central Florida
♢Tencent AI Lab, Bellevue, WA, USA

swcho@knights.ucf.edu {ailabchenli, dyu}@tencent.com {foroosh,feiliu}@cs.ucf.edu

Abstract

Emerged as one of the best performing techniques for extractive summarization, determinantal point processes select the most probable set of sentences to form a summary according to a probability measure defined by modeling sentence prominence and pairwise repulsion. Traditionally, these aspects are modelled using shallow and linguistically informed features, but the rise of deep contextualized representations raises an interesting question of whether, and to what extent, contextualized representations can be used to improve DPP modeling. Our findings suggest that, despite the success of deep representations, it remains necessary to combine them with surface indicators for effective identification of summary sentences.

1 Introduction

Determinantal point processes, shortened as DPP, is one of a number of optimization techniques that perform remarkably well in summarization competitions (Hong et al., 2014). These optimization-based summarization methods include integer linear programming (Gillick and Favre, 2009), minimum dominating set (Shen and Li, 2010), maximizing submodular functions under a budget constraint (Lin and Bilmes, 2010; Yogatama et al., 2015), and DPP (Kulesza and Taskar, 2012). DPP is appealing to extractive summarization, since not only has it demonstrated promising performance on summarizing text/video content (Gong et al., 2014; Zhang et al., 2016; Sharghi et al., 2018), but it has the potential of being combined with deep neural networks for better representation and selection (Gartrell et al., 2018).

The most distinctive characteristic of DPP is its decomposition into the *quality* and *diversity* measures (Kulesza and Taskar, 2012). A *quality* measure is a positive number indicating how important a sentence is to the extractive summary. A *diversity* measure compares a pair of sentences for redundancy. If a sentence is of high quality, any *set* containing it will have a high probability score. If two sentences contain redundant information, they cannot both be included in the summary, thus any *set* containing both of them will have a low probability. DPP focuses on selecting the most probable *set* of sentences to form a summary according to sentence quality and diversity measures.

To better measure quality and diversity aspects, we draw on deep contextualized representations. A number of models have been proposed recently, including ELMo (Peters et al., 2018), BERT (Devlin et al., 2018), XLNet (Yang et al., 2019; Dai et al., 2019), RoBERTa (Liu et al., 2019) and many others. These representations encode a given text into a vector based on left and right context. With carefully designed objectives and billions of words used for pretraining, they have achieved astonishing results in several tasks including predicting entailment relationship, semantic textual similarity, and question answering. We are particularly interested in leveraging BERT for better sentence quality and diversity estimates.

This paper extends on previous work (Cho et al., 2019) by incorporating deep contextualized representations into DPP, with an emphasis on better sentence selection for extractive multi-document summarization. The major research contributions of this work include the following: (*i*) we make a first attempt to combine DPP with BERT representations to measure sentence quality and diversity and report encouraging results on benchmark summarization datasets; (*ii*) our findings suggest that it is best to model sentence *quality*, i.e., how important a sentence is to the summary, by combining semantic representations and surface indicators of the sentence, whereas pairwise sentence *dissimilarity* can be determined by semantic repre-

Proceedings of the 2nd Workshop on New Frontiers in Summarization, pages 98–103
Hong Kong, China, November 4, 2019. ©2019 Association for Computational Linguistics

sentations only; (*iii*) our analysis reveals that combining contextualized representations with surface features (e.g., sentence length, position, centrality, etc) remains necessary, as deep representations, albeit powerful, may not capture domain-specific semantics/knowledge such as word frequency.

2 DPP for Summarization

Determinantal point process (Kulesza and Taskar, 2012) defines a probability measure $\mathcal{P}$ over all subsets ($2^{|\mathcal{Y}|}$) of a ground set containing all document sentences $\mathcal{Y} = \{1, 2, \cdots, \mathsf{N}\}$. Our goal is to identify a most probable subset Y, corresponding to an extractive summary, that achieves the highest probability score. The probability measure $\mathcal{P}$ is defined as

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

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

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 semi-definite (PSD) matrix, known as the L-ensemble; L_{ij} indicates the correlation between sentences i and j; and L_Y is a submatrix of L containing only entries indexed by elements of Y. As illustrated in Eq. (1), the probability of an extractive summary $Y \subseteq \mathcal{Y}$ is thus proportional to the determinant of the matrix L_Y.

Kulesza and Taskar (2012) introduce 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 number indicating the *quality* of a sentence and S_{ij} is a measure of *similarity* between sentences i and j. The q and S model the sentence quality and pairwise similarity respectively and contribute to the L-ensemble matrix. A log-linear model is used to determine sentence quality: $q_i = \exp(\boldsymbol{\theta}^\top \mathbf{f}(i))$, where $\mathbf{f}(i)$ is a feature vector for sentence i and $\boldsymbol{\theta}$ are feature weights to be learned during DPP training. We optimize $\boldsymbol{\theta}$ by maximizing log-likelihood with gradient descent, illustrated as follows:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{m=1}^{M} \log \mathcal{P}(\hat{Y}^{(m)}; L^{(m)}(\boldsymbol{\theta})), \quad (3)$$

$$\nabla_{\boldsymbol{\theta}} = \sum_{m=1}^{M} \sum_{i \in \hat{Y}^{(m)}} \mathbf{f}(i) - \sum_{j} \mathbf{f}(j) K^{(m)}_{jj}, \quad (4)$$

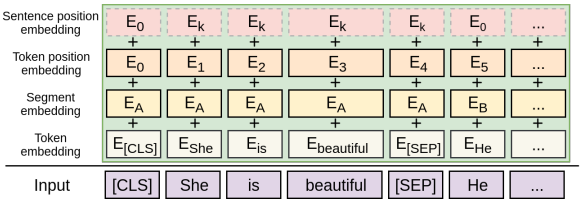

Figure 1: BERT-*sim* and BERT-*imp* utilize embeddings for tokens, segments, token position in a sentence and sentence position in a document. These embeddings are element-wisely added up then fed into the model.

CNN/DM	mean	min	max
train-pos	13.95	1	318
train-neg	21.90	1	337
DUC-04	2.22	1	5
TAC-11	1.67	1	5

Table 1: Position of summary-worthy sentences in a document for single-doc (CNN/DM) and multi-doc datasets (DUC-04, TAC11). 'pos' are summary-worthy document sentences; 'neg' are sentences that are randomly sampled from the same document.

where M is the total number of training instances; $\hat{Y}^{(m)}$ is the ground-truth summary of the m-th instance; $K = L(L + I)^{-1}$ is the kernel matrix and $\mathcal{P}(\hat{Y}^{(m)}; L^{(m)}(\boldsymbol{\theta}))$ is defined by Eq. (1). We refer the reader to (Kulesza and Taskar, 2012) for details on gradient derivation (Eq. (4)). In the following we describe two BERT models to respectively estimate sentence pairwise similarity and importance. The trained models are then plugged into the DPP framework for computing S and q.

2.1 BERT Architecture

We introduce two models that fine-tune the BERT-base architecture (Devlin et al., 2018) to calculate the similarity between a pair of sentences (BERT-*sim*) and learn representations that characterize the importance of a single sentence (BERT-*imp*). Importantly, training instances for both BERT models are derived from *single-document* summarization dataset (Hermann et al., 2015) by Lebanoff et al. (2019), containing a collection of single sentences (or sentence pairs) and their associated labels. During testing, the trained BERT models are applied to single sentences and sentence pairs derived from *multi-document* input to obtain quality and similarity measures.

BERT-*sim* takes as input a pair of sentences and transforms each token in the sentence into an embedding using an embedding layer. They are then passed through the BERT-base architecture to pro-

duce a vector representing the input sentence pair. The vector, denoted by $\mathbf{u} \in \mathbb{R}^d$, is the final hidden state corresponding to the "[CLS]" token (d=768), which is used as the aggregate sequence representation. $\mathbf{u}$ is passed through a feed-forward layer with the same dimension d, followed by a dropout layer, and a final softmax prediction layer to classify whether a pair of sentences contain redundant information or not. Once the model is trained, we can apply it to a pair of sentences i and j to obtain the similarity score S_{ij}.

BERT-*imp* uses a similar architecture to predict if any single sentence is important to the summary. Once the model is trained, we can apply it to the i-th sentence to generate a vector $\mathbf{u}_i$ which is used as the feature representation $\mathbf{f}(i)$ for the i-th sentence when computing q_i.

The embedding layer, illustrated in Fig. 1, consists of several types of embeddings, respectively representing tokens, segments, the token position in a sentence and sentence position within a given document. These embeddings are element-wisely added up then fed to the model. The sentence position embeddings are incorporated in this work to capture the position of a sentence in the article. It is utilized only by BERT-*imp*, as position matters for sentence importance but not quite so for pairwise similarity. As shown in Table 1, positive sentences in the training data (see §3.1) tend to appear at the beginning of an article, consistently more so than negative sentences. Further, ground-truth summary sentences of the DUC and TAC datasets are likely to appear among the first five sentences of an article, indicating position embeddings are crucial for training the BERT-*imp* model.

2.2 DPP Training

DPP training focuses on estimating the weights of features used in $q_i = \exp(\boldsymbol{\theta}^\top \mathbf{f}(i))$, which is a log-linear model used for computing sentence quality. The sentence similarity scores S_{ij} are produced by BERT-*sim*; they do not change during DPP training. We obtain contextualized representations for the i-th sentence, i.e., $\mathbf{f}(i) \in \mathbb{R}^d$, from the penultimate layer ($\mathbf{u}_i$) of BERT-*imp*.

In addition, a number of surface indicators[1], denoted by $\mathbf{v}_i \in \mathbb{R}^{d'}$, are extracted for sentence i. To combine surface indicators and contextualized representations, we concatenate $\mathbf{u}_i$ and $\mathbf{v}_i$ as sentence features. We also take a weighted average[2] of S_{ij} and C_{ij} as an estimate of pairwise sentence similarity, where C_{ij} is the cosine similarity of sentence TF-IDF vectors. DPP training learns feature weights $\boldsymbol{\theta} \in \mathbb{R}^D$, where $D = d + d'$ if the sentence features are concatenated, otherwise $D = d$. DPP is trained on multi-document summarization data with gradient descent (Eq. (4)).

[1]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 abstain from using sophisticated features to avoid model overfitting.

System	DUC-04 R-1	R-2	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
ICSISumm (Gillick and Favre, 2009)	37.31	9.36	13.12
DPP (Kulesza and Taskar, 2012)†	38.10	9.14	13.40
DPP-Caps (Cho et al., 2019)	38.25	9.22	13.40
DPP-Caps-Comb (Cho et al., 2019)	39.35	10.14	14.15
DPP-BERT (ours)	38.14	9.30	13.47
DPP-BERT-Comb 64 (ours)	38.78	9.78	14.04
DPP-BERT-Comb 128 (ours)	**39.05**	**10.23**	**14.35**

Table 2: Results on the DUC-04 dataset evaluated by ROUGE. † indicates our reimplementation of Kulesza and Taskar (2012) system.

3 Experiments

In this section we describe the dataset used to train the BERT-*sim* and BERT-*imp* models, benchmark datasets for multi-document summarization, and experimental settings. Our system shows competitive results comparing to state-of-the-art methods. Example summaries are provided to demonstrate the effectiveness of the proposed method.

3.1 Dataset

CNN / DailyMail This dataset (Hermann et al., 2015) is utilized to train the BERT-*sim* and BERT-*imp* models. For BERT-*sim*, we pair each human summary sentence with its most similar document sentence to create a positive instance; negative instances are randomly sampled sentence pairs. For BERT-*imp*, the most similar document sentence receives a label of 1; randomly sampled sentences are labelled as 0. In total, our training / dev / test sets contain 2,084,798 / 105,936 / 86,144 sentence pairs and the instances are balanced.

[2]The coefficient is set to be 0.9 for both datasets.

DUC/TAC We evaluate our DPP approach (§2) on multi-document summarization datasets including DUC and TAC (Over and Yen, 2004; Dang and Owczarzak, 2008). The task is to generate a summary of 100 words from a collection of news articles. We report ROUGE F-scores (Lin, 2004)[3] on DUC-04 (trained on DUC-03) and TAC-11 (trained on TAC-08/09/10) following standard settings (Hong et al., 2014). Ground-truth extractive summaries used in DPP training are obtained from Cho et al. (2019).

3.2 Experiment Settings

We implement our system using TensorFlow on an NVIDIA 1080Ti GPU. We consider the maximum length of a sentence to be 64 or 128 words. The batch size is 64 for the 64 max sentence length and 32 for 128. We use Adam optimizer (Kingma and Ba, 2015) with the default setting and set learning rate to be 2e-5. We train BERT-*imp* and BERT-*sim* on CNN/DM. The prediction accuracy of BERT-*sim* and BERT-*imp* (with length-128) are respectively 96.11% and 69.05%. Similar results are observed with length-64: 95.79% and 69.63%.

3.3 Summarization Results

We compare our system with strong summarization baselines (Table 2 and 3). *SumBasic* (Vanderwende et al., 2007), *KL-Sum* (Haghighi and Vanderwende, 2009), and *LexRank* (Erkan and Radev, 2004) are extractive approaches; *Opinosis* (Ganesan et al., 2010), *Extract+Rewrite* (Song et al., 2018), and *Pointer-Gen* (See et al., 2017) are abstractive methods; *ICSISumm* (Gillick et al., 2009) is an ILP-based summarization method; and *DPP-Caps-Comb*, *DPP-Caps* are results combining DPP and capsule networks reported by Cho et al. (2019) w/ and w/o using sentence TF-IDF similarity ($C_{i,j}$).

We experiment with variants of our DPP model: *DPP-BERT*, *DPP-BERT-Combined*. The former utilizes the outputs from BERT-*sim* and BERT-*imp* to compute S_{ij} and q_i, whereas the latter combines BERT-*sim* output with sentence TF-IDF similarity ($C_{i,j}$), and concatenates BERT-*imp* features with linguistically informed features.

Our DPP methods outperform both extractive and abstractive baselines, indicating the effectiveness of optimization-based methods for extractive multi-document summarization. Furthermore, we

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

System	TAC-11 R-1	TAC-11 R-2	TAC-11 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, 2012)†	36.95	9.83	13.57
DPP-Caps (Cho et al., 2019)	36.61	9.30	13.09
DPP-Caps-Comb (Cho et al., 2019)	37.30	10.13	13.78
DPP-BERT (ours)	37.04	10.18	13.79
DPP-BERT-Comb 64 (ours)	38.46	10.79	14.45
DPP-BERT-Comb 128 (ours)	**38.59**	**11.06**	**14.65**

Table 3: ROUGE results on the TAC-11 dataset.

observe that *DPP-BERT-Combined* yields the best performance, achieving 10.23% and 11.06% F-scores respectively on DUC-04 and TAC-11. This finding suggests that sentence similarity scores and importance features from the *DPP-BERT* system and TF-IDF based features can complement each other to boost system performance. We conjecture that TF-IDF sentence vectors are effective at representing topical terms (e.g., *3 million*), thus helping DPP better select representative sentences. Another observation is that *DPP-BERT* and *DPP-BERT-Combined* consistently outperform *DPP-Caps* and *DPP-Caps-Comb*, indicating its excellence for DPP-based summarization.

In Table 4 we show example system summaries and a human-written reference summary. *DPP-BERT* and *DPP-BERT-Combined* both are capable of selecting a balanced set of representative and diverse summary sentence from multi-documents. *DPP-BERT-Combined* selects more relevant sentences than *DPP-BERT* comparing to the human summary, leading to better ROUGE scores.

4 Conclusion

In this paper we describe a novel approach using determinantal point processes for extractive multi-document summarization. Our DPP+BERT models harness the power of deep contextualized representations and optimization to achieve outstanding performance on multi-document summarization benchmarks. Our analysis further reveals that, despite the success of deep contextualized representations, it remains necessary to combine them with surface indicators for effective identification of summary-worthy sentences.

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.

DPP-BERT Summary

- The petition is far from the first warning about children using the medicines.
- The FDA will formally consider revising labeling at a meeting scheduled for Oct. 18-19.
- 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.
- Similarly, hydrocodone has never been shown to be safe and effective in children, and its dangers as a powerful and potentially addictive narcotic are clear.

DPP-BERT-Combined Summary

- The U.S. government is warning parents not to give cough and cold medicines to children under 2 without a doctor's order, part of an overall review of the products' safety and effectiveness for youngsters.
- Drug makers on Thursday voluntarily pulled kids' cold medicines off the market less than two weeks after the U.S. government warned of potential health risks to infants.
- 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.
- In high doses, cold medicines can affect the heart's electrical system, leading to arrhythmias.

Table 4: Example system summaries and their human reference summary. Sentences selected by DPP-BERT-Combined are more similar to the human summary than those of DPP-BERT; both include diverse sentences.

Acknowledgments

We are grateful to the anonymous reviewers for their helpful suggestions. This research was supported in part by the National Science Foundation grant IIS-1909603.

References

Sangwoo Cho, Logan Lebanoff, Hassan Foroosh, and Fei Liu. 2019. Improving the similarity measure of determinantal point processes for extractive multi-document summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*.

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

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)*.

Mike Gartrell, Elvis Dohmatob, and Jon Alberdi. 2018. Deep determinantal point processes. *https://arxiv.org/abs/1811.07245*.

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*.

Dan Gillick, Benoit Favre, Dilek Hakkani-Tur, Berndt Bohnet, Yang Liu, and Shasha Xie. 2009. The ICSI/UTD summarization system at TAC 2009. In *Proceedings of TAC*.

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)*.

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)*.

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)*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

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)*.

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*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *https://arxiv.org/pdf/1907.11692.pdf*.

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

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

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)*.

Aidean Sharghi, Ali Borji, Chengtao Li, Tianbao Yang, and Boqing Gong. 2018. Improving sequential determinantal point processes for supervised video summarization. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

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)*.

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.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *https://arxiv.org/abs/1906.08237*.

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)*.

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)*.