

Extractive Summarization Considering Discourse and Coreference Relations based on Heterogeneous Graph

Yin Jou Huang
Kyoto University

huang@nlp.ist.i.kyoto-u.ac.jp

Sadao Kurohashi
Kyoto University

kuro@i.kyoto-u.ac.jp

Abstract

Modeling the relations between text spans in a document is a crucial yet challenging problem for extractive summarization. Various kinds of relations exist among text spans of different granularity, such as discourse relations between elementary discourse units and coreference relations between phrase mentions. In this paper, we propose a heterogeneous graph based model for extractive summarization that incorporates both discourse and coreference relations. The heterogeneous graph contains three types of nodes, each corresponds to text spans of different granularity. Experimental results on a benchmark summarization dataset verify the effectiveness of our proposed method.

1 Introduction

Automatic summarization aims to condense the information of the input document into a shorter summary. The task has two main paradigms: extractive summarization and abstractive summarization. Generating summary sentences from scratch, abstractive summarizers can generate concise and flexible summaries. However, they also suffer from the problem of not being able to reproduce factual details correctly (See et al., 2017). On the other hand, extractive summarization aims to select salient text spans (mostly sentences) from the input document. Compared to abstractive summarizers, extractive summarizers have the advantage of being efficient and factually reliable. In this paper, we will focus on extractive summarization.

For extractive summarization, it is crucial to model the relations between text spans throughout the document. Between text spans of different granularity, there exist many different kinds of relations (Figure 1). For example, coreference relations exist between mention phrases of the same entity, and discourse relations exist between Elementary Discourse Units (EDUs) within a document. Due

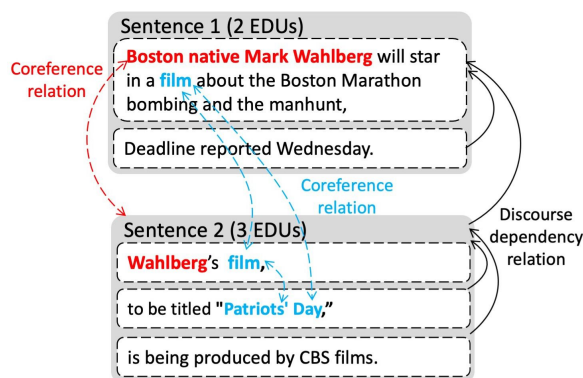


Figure 1: Relations among text spans of different granularity.

to its complex nature, modeling the various relations among text spans of a document remains an open challenge.

To capture inter-sentential relations, some recent works utilize recurrent neural networks (RNNs) or Transformer (Vaswani et al., 2017) based encoders on top of the acquired sentence representations (Cheng and Lapata, 2016; Nallapati et al., 2016; Liu and Lapata, 2019). However, empirical observations show that these sentence-level encoders do not bring much performance gain (Liu and Lapata, 2019). Graph structure is an intuitive way to model long-range dependencies among text spans throughout a document. Early works build connectivity graphs based on content similarity between sentences (Erkan and Radev, 2004; Mihalcea and Tarau, 2004). Some recent works incorporate discourse or coreference relations into the graph structure and utilize graph neural networks (GNNs) to obtain a high-level representation of text spans (Yasunaga et al., 2017; Xu and Durrett, 2019; Xu et al., 2020). Most of these works operate on homogeneous graphs with only one type of nodes, such as Approximate Discourse Graph (ADG) (Christensen et al., 2013) or Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) dependency graph. As illustrated in Figure 1, the various types of relations exist between text spans of different

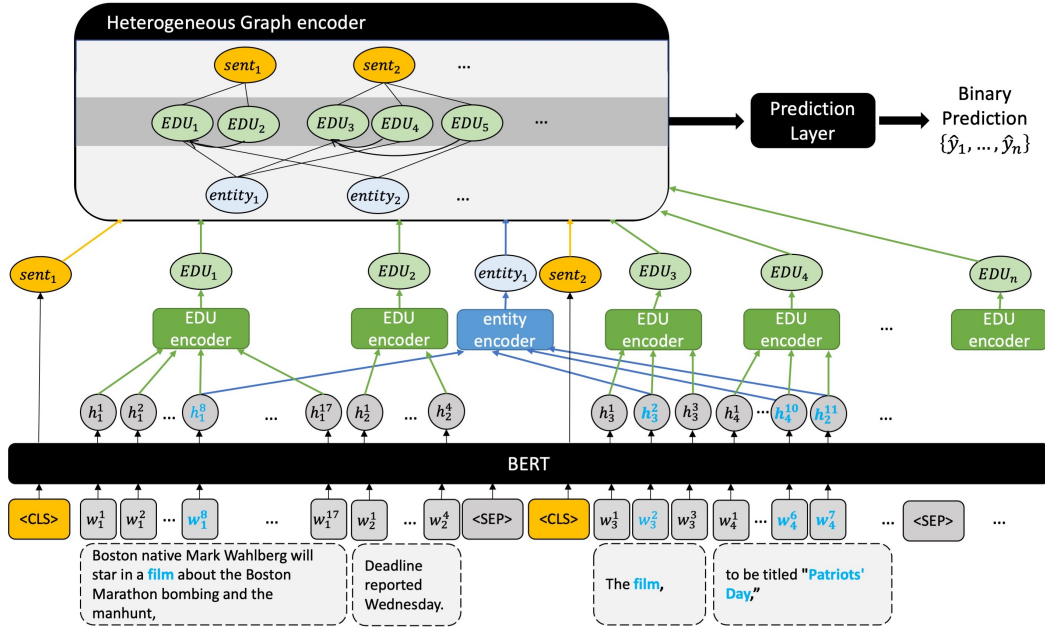


Figure 2: System overview.

granularity. Thus, homogeneous graphs may not be an ideal way to encode the various types of relations between text spans.

In this paper, we propose a novel heterogeneous graph based model for extractive summarization. Heterogeneous graphs are graphs that contain multiple node types and/or multiple edge types, which is in contrast to homogeneous graphs that have only one type of nodes and one type of edge. Heterogeneous graphs have been widely studied and applied to model data structures such as citation networks (Yu et al., 2012), recommendation systems (Dong et al., 2012), etc. In this work, we use heterogeneous graph to model the document structure with three types of nodes of different granularity: sentence nodes, EDU nodes, and entity nodes.

We also try to encode both discourse and coreference relations into the graph structure. We encode the discourse relations with the edges between EDU nodes. As for the coreference relations, edges between EDU nodes and entity nodes are introduced. Instead of extracting salient sentences like most existing extractive summarizers, our model extracts salient EDUs. To identify salient EDUs in a certain sentence, we add edges between sentence nodes and their constituent EDU nodes. To the best of our knowledge, we are the first to utilize heterogeneous graph to incorporate multiple types of relations simultaneously for extractive summarization.

2 Proposed Method

Given an input document D with n EDUs $\{d_1, d_2, \dots, d_n\}$, we formulate extractive summarization as a sequence labeling problem. The model predicts a sequence of binary labels $Y = \{y_1, y_2, \dots, y_n\}$, where $y_i = 1$ indicates that the i^{th} EDU should be included in the summary.

Figure 2 provides an overview of our proposed model. First, a BERT encoder is used to embed the input document D . With the EDU and entity encoders, we acquire the initial node representation of the heterogeneous graph. We then apply a heterogeneous graph encoder to obtain high-level node representations. Finally, we make predictions based on the EDU node representations.

2.1 Heterogeneous Graph Construction

We represent each input document D with a heterogeneous graph $G = \{V, E\}$, where V and E are the set of nodes and edges, respectively.

Given document D with m sentences $\{s_1, \dots, s_m\}$, we first segment the sentences into n sub-sentential EDUs $\{d_1, \dots, d_n\}$ and perform RST discourse parsing to identify the relations between the EDUs. In addition, we perform coreference resolution to identify the mentions and the coreference relations between them. The mentions in D are then clustered into k entities $\{e_1, \dots, e_k\}$, with each entity e_i representing a cluster of mentions among which coreference relations holds.

The set of nodes $V = V_s \cup V_d \cup V_e$ consists of m

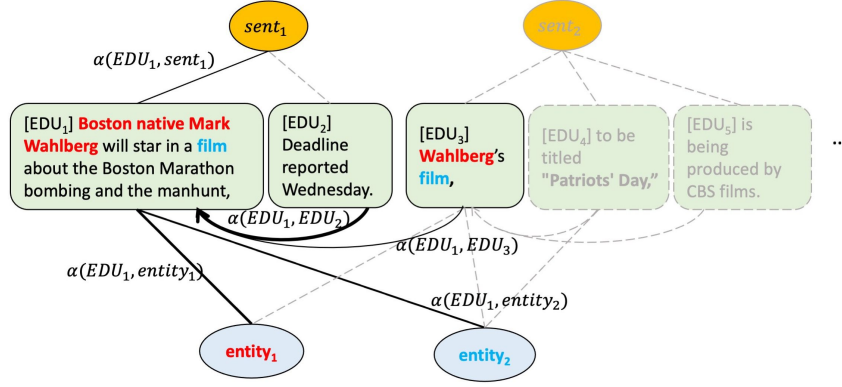


Figure 3: Graph attention mechanism.

sentence nodes V_s , n EDU nodes V_d , and k entity nodes V_e . There are three types of edges in E . First, we use edges between EDU nodes to represent the discourse structure of the document. Similar to Xu et al. (2020), we derived the discourse dependency links between EDU nodes based on the RST tree of the document. The discourse dependency links are directional, which capture the dependency relations going from satellite to nucleus EDUs. Second, we use edges between EDU nodes and entity nodes to embed the coreference relations. If EDU d_i contains a mention of entity e_j , then we add an undirected edge (d_i, e_j) to E . In this way, each entity indirectly connects all EDUs with mentions of the entity. Third, we also link each sentence node to its constituent EDU nodes undirectionally.

The proposed heterogeneous document graph enables us to simultaneously model various relations between different sizes of text spans: sentence, EDU, entity phrase efficiently.

2.2 Graph Node Initialization

Following the settings in Liu and Lapata (2019), we utilize pretrained BERT (Devlin et al., 2019) to encode the input document D . We insert the $\langle \text{CLS} \rangle$ and $\langle \text{SEP} \rangle$ special tokens to the beginning and the end of each sentence s_i , respectively. With the BERT output vectors, we acquire the initial representations of each node in V as follows:

Sentence Representations

For each sentence node s_i , we take the BERT output vector of the $\langle \text{CLS} \rangle$ token before s_i as the sentence node representation $sent_i$.

EDU Representations

We use a self-attention based EDU encoder to encode each EDU node d_i . Given an EDU d_i with tokens $\{w_i^j\}$, we obtain its node representation EDU_i by taking self-attention on the BERT output

vectors $\{h_i^j\}$ of the tokens:

$$\alpha_{ij} = v_2 \text{ReLU}(W_1 h_i^j + b_1) \quad (1)$$

$$a_{ij} = \text{softmax}_j(\alpha_{i1}, \alpha_{i2}, \dots) \quad (2)$$

$$EDU_i = \sum_j a_{ij} h_i^j \quad (3)$$

Entity Representations

The structure of the entity encoder is identical to the EDU encoder. For each entity e_i , we consider all mentions of it. By taking self-attention among the BERT output vectors which correspond to tokens of these mentions, we can acquire the entity representation $entity_i$.

2.3 Heterogeneous Graph Encoder

We initialize the representation of each node in G with the sentence representations ($sent_i$), EDU representations (EDU_i), and entity representations ($entity_i$) acquired in section 2.2.

We apply graph attention networks (GAT) (Veličković et al., 2018) to update the node representations in G . For each iteration, we update the representation h_i of node i with the representations of its neighbors $\{h_j\}$ based on the attention weights α_{ij} :

$$\alpha_{ij} = \text{LeakyReLU}(W_a[W_q h_i; W_k h_j]) \quad (4)$$

$$a_{ij} = \text{softmax}_j(\alpha_{i1}, \alpha_{i2}, \dots) \quad (5)$$

$$h_i \leftarrow W_t(\sigma(\sum_j a_{ij} W_v h_j) + h_i) \quad (6)$$

An example of the graph attention mechanism is illustrated in Figure 3, where the subgraph around node EDU_1 is highlighted. EDU_1 has five neighbors: a sentence node ($sent_1$), two EDU nodes (EDU_2, EDU_3), and two entity nodes ($entity_1, entity_2$.) We first calculate the attention weights α across the five neighbors of EDU_1 using equation

4, and update the node representation of EDU_1 accordingly.

Although a single GAT network only considers the first-degree neighbors, we can obtain a higher-level representation for each node in G by updating the node representations for several iterations.

2.4 Prediction Layer

We feed the final representation of the EDU nodes (EDU_i) to the prediction layer with sigmoid activation to predict binary labels:

$$\hat{y}_i = \sigma(W_p EDU_i + b_p) \quad (7)$$

The training loss of the model is the binary cross-entropy loss against the oracle extraction labels.

3 Experiment

3.1 Dataset

We evaluated our proposed model on the benchmark CNN/DailyMail dataset (non-anonymized version) (Hermann et al., 2015). We used the standard dataset split, which contains 287,227 / 13,368 / 11,490 documents for training, validation, and test split, respectively.

We used the Stanford CoreNLP (Manning et al., 2014) to split sentences. Further, we used the RST discourse parser proposed by Ji and Eisenstein (2014) for both discourse segmentation and discourse parsing. For coreference resolution, we used the spanBERT-based (Joshi et al., 2020) version of the end-to-end coreference resolver proposed by Lee et al. (2017).

Since the CNN/DailyMail dataset only contains abstractive gold summaries, we have to construct oracle labels heuristically. We obtained the oracle labels on EDU-level with the heuristic algorithm based on ROUGE (Lin, 2004), similar to the one in Liu and Lapata (2019). For each document, we selected up to 5 EDUs.

3.2 Experimental Settings

We used the base model of Longformer (Beltagy et al., 2020) to encode the input document. The length of each document is truncated to 1024 BPEs. The hidden size of the EDU encoder and the entity encoder is 128. Based on the evaluation losses on the validation set, we set the number of iterations of the GAT layer to 3. Also, the number of attention heads is set to 8, with each head having a hidden size 64.

Model	R-1	R-2	R-L
LEAD-3	40.34	17.70	36.57
Oracle(sent)	52.59	31.24	48.87
Oracle(EDU)*	55.96	34.64	53.26
BanditSum (Dong et al., 2018)	41.50	18.70	37.60
NEUSUM (Zhou et al., 2018)	41.59	19.01	37.98
HIBERT (Zhang et al., 2019)	42.37	19.95	38.83
HSG (Wang et al., 2020)	42.95	19.76	39.23
BertSum (sent) (Liu and Lapata, 2019)	43.25	20.24	39.63
BertSum (EDU)*	42.73	20.03	40.16
DiscoBERT* (Xu et al., 2020)	43.77	20.85	40.67
Proposed*	43.61	20.81	41.12

Table 1: Results on the test set of CNN/DailyMail dataset. Models with * superscript are EDU-based extractive models.

During training, we used a batch size of 32. We used Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and followed the learning rate scheduling in Vaswani et al. (2017) with warm-up of 4000 steps. All models are trained for 50000 steps. We selected the top-3 checkpoints based on the evaluation losses on the validation set and report the average scores of them on the test set.

3.3 Results and Analysis

Table 1 shows the results on CNN/DailyMail dataset. The first part contains the **LEAD-3** baseline and **Oracle** upper bounds. The second part of the table includes other sentence based extractive models, and the third part includes other EDU based extractive models. In the last row of the table, we present the evaluation scores of our proposed model.

As Table 1 shows, our proposed model outperforms the **BertSum(EDU)** baseline by a significant margin (0.88/0.78/0.96 on F₁ of R-1/R-2/R-L). Our proposed model also outperforms the **BertSum(sent)** model and other sentence based extractive summarization baseline models. The proposed model is comparable to the state-of-the-art EDU-extraction model **DiscoBERT** in R-1 and R-2 metrics, and outperforms it in R-L metrics. **DiscoBERT** incorporates a strict RST-based rule during both oracle label construction and post-processing stages to ensure discourse consistency. Since the purpose of this paper is to propose a heterogeneous graph based method for modeling text span relations, we will leave the question of discourse consistency to future work.

Ablation Study

We conduct an ablation study on the components of our proposed model (Table 3). First, we remove the RST dependency edges between EDU nodes (**-discourse**). Next, we remove the corefer-

EDUs with coreferent entity	Rank _{coref}	Rank _{w/o coref}	Oracle Label
Mexican state oil company Pemex said 45 workers were injured ...	1	2	1
... two of them are in serious condition	2	1	0
Kim was accused of stabbing U.S. ambassador Mark Lippert ...	1	3	1
Before Lippert was supposed to give a speech, the attacker slashed him in the face a jaw	3	1	0
Kim stabbed Lippert with a 10-inch knife	2	2	0

Table 2: Qualitative studies on CNN/DailyMail dataset.

Model	R-1 (Δ R-1)	R-2 (Δ R-2)	R-L (Δ R-L)
Proposed	43.61	20.67	40.95
- discourse	43.42(-0.19)	20.54(-0.13)	40.77(-0.18)
- coref	43.51(-0.10)	20.61(-0.06)	40.85(-0.10)
- sent	43.57(-0.04)	20.66(-0.01)	40.90(-0.05)

Table 3: Ablation studies on CNN/DailyMail dataset.

ential edges between EDU nodes and entity nodes (**-coref**). The results of the ablation study show that discourse information plays an important role in our proposed model, while adding coreference information also gives a gain in performance. We also try to remove the edges between sentence nodes and their constituent EDU nodes (**-sent**). However, linking the sentence and EDU nodes does not seem to have a significant impact on model performance.

Qualitative Analysis

We also conduct a qualitative analysis of the proposed model. The effectiveness of discourse relations is more straight-forward and widely studied in previous research. Thus, we focus on the analysis of the role of coreference information in our proposed summarization model.

In the heterogeneous document graph, EDUs containing the same entity phrase are indirectly connected through the node of the given entity. By analyzing the output of the full proposed model and the model without coreference information (**-coref**), we found that the models rank the importance of coreferent EDUs differently. Table 2 indicates a common pattern of the improved cases by incorporating coreference information. The table shows examples of coreferent EDUs and the ranking of their likelihood scores to be included in the summary. Comparing the EDU ranking of the full model (**Rank_{coref}**) and the model without coreference information (**Rank_{w/o coref}**), we argue that the model with coreference information is better in discriminating the important EDUs among all EDUs sharing the same entity.

4 Related Work

Graph based Summarization

Graph based summarization models have been broadly explored. Early works build connectivity

document graphs based on inter-sentential similarity (Erkan and Radev, 2004; Mihalcea and Tarau, 2004). With the promising results of graph neural networks (GNNs)(Kipf and Welling, 2017; Veličković et al., 2018), some recent works utilize GNN to incorporate external knowledge into the model. For instance, Yasunaga et al. (2017) utilizes a sentence-level ADG graph to model discourse and coreference relations. Some works convert the RST tree of the input document into dependency form in either sentence or EDU level (Xu and Durrett, 2019; Xu et al., 2020). Most of these models operate on homogeneous graphs with only one type of node. One of the major disadvantages of homogeneous graphs is that they can only embed one relation type in a single graph, since there is only one type of node and one type of edge.

Fewer summarization models operate on heterogeneous graphs with different types of nodes. Wei (2012) introduces a heterogeneous graph of sentence, word, and topic nodes, and Wang et al. (2020) also utilizes a heterogeneous graph of sentence and word nodes. However, neither of the above works incorporates external knowledge into the graph.

EDU based Extractive Summarization

Li et al. (2016) illustrates the potential of using EDU as the extraction unit for summarization. Xu et al. (2020) also introduces an end-to-end EDU based extractive summarization model. By using a heuristic based on RST dependency structure, they enhanced the grammaticality and discourse consistency of the extracted summary.

5 Conclusion

In this paper, we proposed a novel heterogeneous graph based model for extractive summarization. By introducing nodes of different granularity, the heterogeneous graph has the capacity to embed various types of relations between text spans. Experiments on CNN/DailyMail benchmark dataset illustrated the effectiveness of our proposed method.

References

- Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.
- Jianpeng Cheng and Mirella Lapata. 2016. **Neural summarization by extracting sentences and words**. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany. Association for Computational Linguistics.
- Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2013. **Towards coherent multi-document summarization**. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1163–1173, Atlanta, Georgia. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. **BERT: Pre-training of deep bidirectional transformers for language understanding**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Y. Dong, J. Tang, S. Wu, J. Tian, N. V. Chawla, J. Rao, and H. Cao. 2012. **Link prediction and recommendation across heterogeneous social networks**. In *2012 IEEE 12th International Conference on Data Mining*, pages 181–190.
- Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. 2018. **Bandit-Sum: Extractive summarization as a contextual bandit**. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3739–3748, Brussels, Belgium. Association for Computational Linguistics.
- Günes Erkan and Dragomir R. Radev. 2004. **Lexrank: Graph-based lexical centrality as salience in text summarization**. *J. Artif. Intell. Res. (JAIR)*, 22:457–479.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. **Teaching machines to read and comprehend**. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.
- Yangfeng Ji and Jacob Eisenstein. 2014. **Representation learning for text-level discourse parsing**. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13–24, Baltimore, Maryland. Association for Computational Linguistics.
- Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. **SpanBERT: Improving pre-training by representing and predicting spans**. *Transactions of the Association for Computational Linguistics*, 8:64–77.
- Thomas N. Kipf and Max Welling. 2017. **Semi-supervised classification with graph convolutional networks**. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.
- Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. **End-to-end neural coreference resolution**. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.
- Junyi Jessy Li, Kapil Thadani, and Amanda Stent. 2016. **The role of discourse units in near-extractive summarization**. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 137–147, Los Angeles. Association for Computational Linguistics.
- Chin-Yew Lin. 2004. **ROUGE: A package for automatic evaluation of summaries**. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Yang Liu and Mirella Lapata. 2019. **Text summarization with pretrained encoders**. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.
- William C Mann and Sandra A Thompson. 1988. **Rhetorical structure theory: Toward a functional theory of text organization**. *Text*, 8(3):243–281.
- Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. **The Stanford CoreNLP natural language processing toolkit**. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.
- R. Mihalcea and P. Tarau. 2004. **TextRank: Bringing order into texts**. In *Proceedings of EMNLP-04 and the 2004 Conference on Empirical Methods in Natural Language Processing*.
- Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2016. **Summarunner: A recurrent neural network based sequence model for extractive summarization of documents**.
- Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. **Get to the point: Summarization with pointer-generator networks**. In *Proceedings of the 55th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Linguistics (Volume 1: Long Papers), pages 654–663, Melbourne, Australia. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. [Graph attention networks](#). In *International Conference on Learning Representations*.

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. [Heterogeneous graph neural networks for extractive document summarization](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6209–6219, Online. Association for Computational Linguistics.

Y. Wei. 2012. Document summarization method based on heterogeneous graph. In *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery*, pages 1285–1289.

Jiacheng Xu and Greg Durrett. 2019. [Neural extractive text summarization with syntactic compression](#).

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. [Discourse-aware neural extractive text summarization](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5021–5031, Online. Association for Computational Linguistics.

Michihiro Yasunaga, Rui Zhang, Kshitij Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. [Graph-based neural multi-document summarization](#). In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 452–462, Vancouver, Canada. Association for Computational Linguistics.

Xiao Yu, Quanquan Gu, Mianwei Zhou, and Jiawei Han. 2012. [Citation Prediction in Heterogeneous Bibliographic Networks](#), pages 1119–1130.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019. [HiBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5059–5069, Florence, Italy. Association for Computational Linguistics.

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 *Proceedings of the 56th Annual Meeting of the Association for Computational*