Enhancing Scientific Document Summarization with Research Community Perspective and Background Knowledge

Sudipta Singha Roy, Robert E. Mercer
University of Western Ontario
London, Ontario, Canada
ssinghar@uwo.ca. mercer@csd.uwo.ca

Abstract

Scientific paper summarization has been the focus of much recent research. Unlike previous research which summarizes only the paper in question, or which summarizes the paper and the papers that it references, or which summarizes the paper and the citing sentences from the papers that cite it, this work puts all three of these summarization techniques together. To accomplish this, we have, by utilizing the citation network, introduced a corpus for scientific document summarization that provides information about the document being summarized, the papers referenced by it, as well as the papers that have cited it. The proposed summarizer model utilizes the referenced articles as background information and the citing articles to capture the impact of the scientific document on the research community. Another aspect of the proposed model is its ability to generate both the extractive and abstractive summaries in parallel. The parallel training helps the counterparts to improve their individual performance. Results have shown that the summaries are of high quality when considering the standard metrics.

Keywords: Scientific Document Summarization, Citation Network, Heterogeneous Graph Neural Network

1. Introduction

Text summarization represents an intricate procedure that entails the automatic condensation of a document, all the while preserving a succinct and coherent rendition of its content. In contrast to the widespread utilization of neural text summarization systems for brief texts (Nallapati et al., 2016; Zhong et al., 2019), their application to longer documents, such as scholarly research publications, has not been markedly prevalent. In the context of summarizing scientific manuscripts, the prevailing method typically involves the selective extraction of content solely from the abstract, introduction, and conclusion segments within the target articles (Yasunaga et al., 2019).

Scientific publications are characterized by their length, complex concepts, and domain-specific knowledge. They follow a structured format with sections, and include citations to previous works serving to explain the subject matter to knowledgeable readers while meeting publisher-imposed page limits. Furnishing summarization models with this background information is crucial for enhancing summary quality (An et al., 2021). Considering this fact, An et al. (2021) utilized the citation network to incorporate the background information when summarizing scientific articles. As a consequence, summarizing scientific articles presents a more daunting task than for other document types.

Moreover, there exists a latent dimension to the impact of any given scientific article at the point of its initial publication, which may become apparent only in subsequent studies by other researchers. While a paper's abstract provides a valuable snapshot of the content as envisioned by the authors, it may fall short in capturing the genuine influence that the paper might wield within its domain over time. This influence has the potential to evolve and assume different dimensions as it reverberates throughout the research community (Yasunaga et al., 2019). For instance, we can examine the abstract presented by Bergsma and Lin (2006):

> We present an approach to pronoun resolution based on syntactic paths. Through a simple bootstrapping procedure, we learn the likelihood of coreference between a pronoun and a candidate noun based on the path in the parse tree between the two entities. This path information enables us to handle previously challenging resolution instances, and also robustly addresses traditional syntactic coreference constraints. Highly coreferent paths also allow mining of precise probabilistic gender/number information. We combine statistical knowledge with well known features in a Support Vector Machine pronoun resolution classifier. Significant gains in performance are observed on several datasets.

This abstract provides a glimpse into the methodologies employed by the authors, whereas the citations underscore the significance of the corpus it presents. For instance:

> For the gender task that we study in our experiments, we acquire class instances by filtering the dataset of nouns and their gen-

LREC-COLING 2024, pages 6048–6058

ders created by Bergsma and Lin (2006). (Bergsma and Van Durme, 2013)

Jaidka et al. (2019)) have discerned this absent facet within the realm of scientific document summarization and have undertaken its remediation by introducing a collaborative endeavour. This endeavour is designed to generate summaries that not only encapsulate the content within the document's body but also encompass the broader perspective of the research community regarding these documents' evolution over time.

There has not been a single concerted effort to amalgamate these two approaches which would entail developing a summarization model that not only assimilates the content of the source document and its contextual background but also possesses the capability to gauge the article's influence on its respective academic community through an examination of the citations it has garnered. Considering this fact, in this work, we have introduced a standalone summarizer model which provides an enriched summary of any scientific document combining the knowledge of the articles referenced in the body of the considered document plus the summary of the citation statements made on that particular article in other works together with a summary of the considered article. In pursuit of this goal, we have introduced a corpus for scientific document summarization that leverages the citation network. This corpus furnishes comprehensive information encompassing the document under scrutiny, the papers referenced within it, and the papers that have subsequently cited it. This corpus is formed from a subset of the SSN corpus (An et al., 2021).

Another aspect of this work is that the introduced summarizer model has the ability to produce both the extractive and abstractive summaries in parallel. The rationale behind generating these two summaries in parallel lies in the reciprocal enhancement that occurs during the creation of each summary. The extractive summarizer represents a fusion of the heterogeneous graph-based neural network (Wang et al., 2020a) and the graph attention network (Welling and Kipf, 2016) and the abstractive summarizer is founded on a Longformer (Beltagy et al., 2020) decoder architecture. These two summarizer units establish a bidirectional information exchange by transmitting supplementary control signals to each other through the loss function. This coordinated approach ensures the concurrent generation of high-quality abstractive and extractive summaries. Prior to utilizing these two summarization units, the considered article is segmented using the segmentation technique proposed by Xing et al. (2020) and for each segment it leverages the citation graph to incorporate background information. Subsequently, employing an hierarchical structure, the summaries of the segments are accumulated, the summary of the citing statements are concatenated, and applying the summarizer unit over these intermediate summaries the final summary is generated. Our contributions can be succinctly summarized as follows:

- We have developed a corpus, utilizing the citation network, for scientific document summarization containing 10k research articles. As per our knowledge, this is the first corpus curating the referenced and citing sides of the citation network for this task.
- We have developed a standalone model combing segmentation and summarization techniques that has the ability to gather background information from the referenced articles and reflect the impact of the work on the corresponding research community considering the citations made on it while generating the summaries of the scientific document.
- The model has the capability to produce both the extractive and abstractive summaries in parallel. This parallel training of these two units allow each other to improve their individual performance.

2. Related Work

In the wake of the remarkable progress in neural network technology, a number of noteworthy research endeavours have emerged in recent years that focus on the generation of extractive summaries (Yasunaga et al., 2019) as well as abstractive summaries (Yu et al., 2020; Zhang et al., 2019) in the realm of scientific document summarization (Cohan et al., 2018; El-Kassas et al., 2021; Zhang et al., 2022; Altmami and Menai, 2022).

Extractive summarizers identify pivotal sentences from the source document to form the summary; however, they tend to lack the coherent flow of information. Inceptive studies (Erkan and Radev, 2004; Mihalcea and Tarau, 2004) employ cosine similarity measurements between sentences for constructing a graph that encapsulates inter-sentence correlations. Certain contemporary research endeavours (e.g., (Cohan and Goharian, 2018; Yasunaga et al., 2017)) have incorporated discourse-related information from the articles in conjunction with inter-sentence correlations to formulate graphs and subsequently generate document summaries. Li et al. (2020) fine-tuned T5 and integrated an extractive summarizer using Graph Convolutional Networks (GCN) for the purpose of generating summaries from extensive scientific documents. Wang et al. (2020a) have employed supplementary semantic units in a graph neural network (GNN) to establish intricate relationships between sentences while designing their

extractive summarizer. Xie et al. (2022) have introduced an extractive summarizer model for long documents (GRETEL) utilizing a graph contrastive topic model and a pre-trained language model. Cho et al. (2022) have introduced a model (Lodoss) which segments the document and extracts the important sentences simultaneously.

Abstractive summarization lays significant emphasis on formulating a generalized summary, often requiring the utilization of sophisticated language generation models. These models are commonly built upon sequence-to-sequence (seq2seq) architectures, wherein the source document is treated as one sequence, while its corresponding summary is considered another. Cohan et al. (2018) pioneered the development of the initial abstractive summarizer designed for lengthy scientific articles. Their approach incorporates an hierarchical encoder and a discourse-aware attentive decoder to accomplish this task. Mishra et al. (2022) implemented a citation contextualization method to extract distinct and pertinent sentences from the document. Subsequently, they employed a multi-objective clustering approach to generate the final summaries. Liu and Lapata (2019) harnessed the encoder-decoder framework of BERT, enabling their model BERTSUMABS to generate abstractive summaries. Wang et al. (2020b) entailed the independent extraction of latent topics from the input text, aiming to capture the underlying themes or concepts within the document. Subsequently, these extracted latent topics are employed to augment the performance of the summarizer. Yu et al. (2020) utilized the guidance of an extractive summarizer to enhance the performance of their abstractive summarizer (DimSum). It employs BART (Lewis et al., 2020) as the foundation for its abstractive summarizer. The amalgamation of loss functions from both the extractive and abstractive summarizers contributes to the model's ability to generate improved lay summaries from scientific documents. Gupta et al. (2022) employed both BERT and graph-based methodologies in their work on biomedical document summarization. PageSum (Liu et al., 2022) reduces memory overhead by treating the input document as a collection of pages based on locality. Each page is independently encoded by the abstractive model's encoder and the decoder generates local predictions for each page and assigns confidence scores to these predictions. HierGNN (Qiu and Cohen, 2022) is a neural encoder with reasoning capabilities, making it compatible for integration into various seq2seq neural summarization models.

A citation network has two sides: the articles being referenced in the considered literature, and the articles that have cited the considered article. To incorporate the information from the referenced articles while summarizing scientific documents, An et al. (2021) introduced a substantial corpus, denoted as SSN, comprising 141,000 research papers interconnected through a citation network. Additionally, they presented a graph-based summarization model called CGSUM to extract information from both the source document and the citing texts, enhancing its summarization capabilities. Yasunaga et al. (2019) introduced a corpus (CL-SciSumNet) comprising 1000 research articles with the citations made on them. The intention of their work is to generate summaries that also portray the contribution of this work on the research community by means of accumulating the citing statements. However, as per our knowledge, there is no work yet available that combines the information from the referenced articles to grasp the background knowledge, and at the same time, the impact of the work by analyzing the citations made on the considered article when generating the summary. Filling this gap has been the motivation of our work presented here.

3. Corpus Creation

Summarizing scientific literature is complex due to the need for contextual background knowledge, including references. Summarizer models require information from referenced articles. Additionally, assessing an article's true impact often requires analyzing citing statements. The SSN corpus offers background information from referenced articles, and the CL-SciSumNet corpus provides citing statements. However, there's no corpus connecting both facets of the citation network.

Considering these factors, we've introduced a corpus tailored for scientific document summarization. This corpus covers both sides of the citation network: the referenced articles and the citing statements. Our corpus is, in part, a subset of the SSN corpus. While the SSN corpus contains background references, it lacks citing statements. To address this, we've enhanced our corpus by adding citing statements from citing papers to bridge this gap and build a more comprehensive corpus.

To create our corpus, we used the citation network to identify citing papers. We then manually extracted statements referencing the cited article from these citing papers. The SSN corpus, with its 141K articles, is quite extensive, so we selected a random subset of 10,000 papers for summarization. These papers have word lengths between 1,000 and 3,500 words, excluding background or related work sections. We deliberately chose this word length range to accommodate both the document and its citing statements within the Longformer's 4,096 token intake limit. In the papers earmarked for summarization, background and related work

sections were removed. The dataset is partitioned into three subsets: 8,000 articles for training, 1,000 for validation, and another 1,000 for testing.

We have categorized the intentions expressed by citations into three classes: positive, neutral, or negative. For each paper, we have selected a maximum of 20 citation statements from each of these categories. Notably, negative citations are less prevalent, so for papers with limited negative citing sentences, we prioritized selecting more neutral and positive ones. To perform this categorization, we have experimented with various BERT-based models and ultimately fine-tuned RoBERTa (Liu et al., 2019) on the Citation Sentiment Corpus (CSC) (Athar, 2011) and used this model to classify all the curated citation statements (Kundu, 2023).

To create summaries that amalgamate the perspectives of both the authors and the broader research community, we took a multi-step approach. Initially, we provided the abstracts of the research papers along with their corresponding citation statements to a fine-tuned T5 model (Raffel et al., 2020). This model had been trained on the CL-SciSumm corpus. It generated five different summaries for each document. Subsequently, these five summaries for each document were fed into a pre-trained Longformer architecture. This process produced five vector representations. To determine the most suitable summary, we compared these vector representations against the reference summary using cosine similarity. The summary with the highest cosine similarity to the reference summary was selected as our T5-Generated Summary, thereby reflecting a synthesis of both the authors' viewpoints and the broader research community's perspectives. To capture the background information we have used the citation network used directly in the SSN corpus. The maximum length of the summaries has been set to 500 tokens. For cleaning the equations and other unnecessary symbols, we have used the regex commands used in Singha Roy and Mercer (2022).

To validate the quality of the proposed corpus, we have performed an analysis on a statistically representative sample of the corpus (95% confidence, and 3% error margin) with three human annotators' assistance. From the pool of 10,000 summarization samples, 400 were chosen randomly and annotated by three annotators for this statistical analysis. Each annotator assessed whether the T5-generated summaries capture the same information as the abstract plus the citing statements. Annotator one said that 374 samples did, annotator two, 368 and annotator three, 371. Annotators one and two agreed that 368 samples compare correctly and 16 do not giving a Cohen's $\kappa = 0.89$. Between annotators two and three the agreement is with 396 samples (368 are correctly summarized and 28 are not) with $\kappa = 0.93$. Between annotators one and three the agreement is found for 398 summaries (370 correctly summarized and 27 not) with $\kappa = 0.94$. The approach for the corpus creation and its validation is shown as a flowchart in Figure 1.

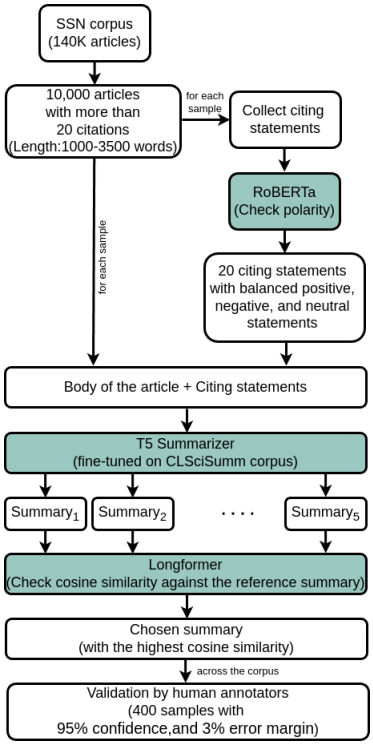

Figure 1: Flowchart of the corpus creation

4. Methodology

This section commences with the problem formulation, outlining how the summarization task of the considered document is enriched by utilizing the information contained in the network of referenced and citing papers. Then, the architecture of the proposed model is discussed.

4.1. Problem Formulation

Scientific papers possess a distinct attribute characterized by the presence of the citation relationship (referring to and referred to) among papers and the logical coherence in their content. Figure 2(a) visualizes this relationship augmented with the ideas of segmenting the considered paper and accumulating only the relevant sentences in the citing papers, both aspects which will be discussed later. These relationships will be used to enhance the effectiveness of summarization tasks in this domain.

To leverage this interconnected nature of scientific literature, we have utilized the concept of a citation graph. Description of this graph and its subgraphs will be used to describe how the model uses

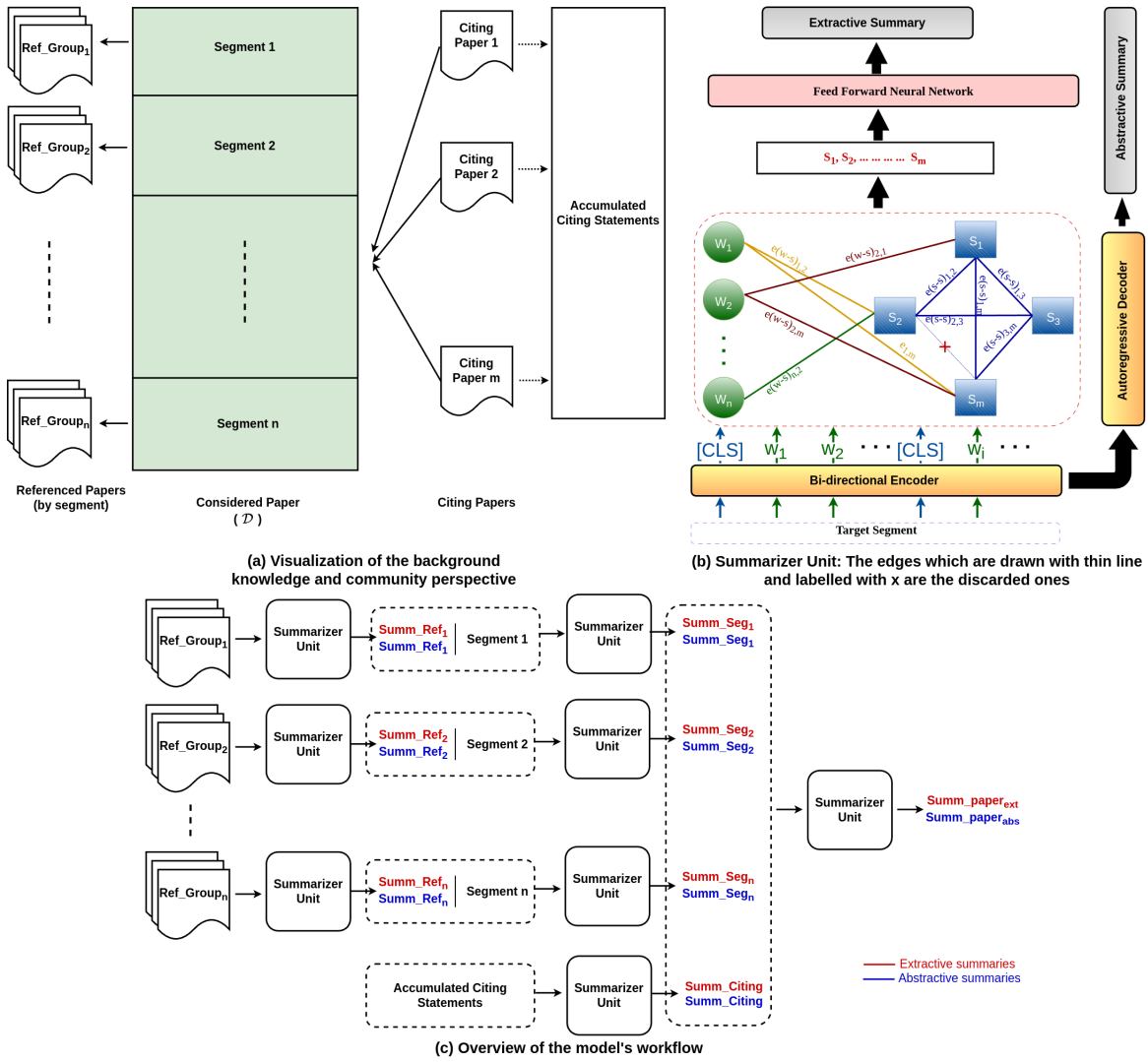

Figure 2: Architecture and workflow of the proposed model.

various portions of it. For the citation graph on the whole dataset $G = (V, E)$, each node $v \in V$ symbolizes a scientific article and each edge $e_{i,j} \in E$ portrays the relationship between articles represented by v_i and v_j. In this graph, the background knowledge for a scientific article v_i, (the papers to the left of the considered paper, $\mathcal{D}$, in Figure 2(a)), is represented by the subgraph G_i^{ref} which contains the relation between v_i and V_i^{ref} which is the set of the articles being referenced by v_i. This is further refined by another characteristic of the scientific article, the structured representation of its information (Cho et al., 2022). To preserve this structure, we have applied the segmentation approach used in Xing et al. (2020). Our work applies this segmentation on document $\mathcal{D}$, the scientific article being considered, to define the citation subgraph $G_i^{Seg_p}$ for each segment $Seg_p, p = 1, ..., n$ in $\mathcal{D}$ to accumulate the background information for all segments.

To define the second subgraph G_i^{citing}, we accumulate the citing statements referring to v_i (see Figure 2(a)) and use this as Seg^{citing}.

4.2. Model Architecture

Our proposed model operates concurrently on the segmentation and summarization tasks, enabling the acquisition of robust sentence representations. The model architecture is portrayed in Figure 2. Initially, the document is segmented into sections using the segmentation model introduced by Xing et al. (2020). This segmentation utilizes the word embeddings from Longformer as input and applies attentive Bi-LSTM on top of it to get the sentence representations. Another sentence representation is

generated from pre-trained Longformer and these two features are concatenated together. This concatenated feature vector is then fed to the following Bi-LSTM layer which predicts the section boundaries. This segmentation problem is formulated as a binary classification problem. To label each article, the sentence starting each segment of the article is labelled with 1 and all others with 0. This segmentation model is optimized with the binary cross entropy loss function (Eq. 1 where k is the number of sentences in the document).

$$\mathcal{L}_{seg} = -\sum_{i=1}^{k-1}[y_i \log p_i + (1-y_i)\log(1-p_i)] \quad (1)$$

After the segmentation is completed, the citation graph is utilized to aggregate the abstracts of the articles referenced by each segment p. For the considered document $\mathcal{D}$, represented by v_i in the citation graph, these articles are represented by the nodes in $G_i^{Seg_p}$. These groups of abstracts are then fed to the subsequent summarizer unit per segment.

The summarizer unit has two components: extractive and abstractive summarizer units. The architectural overview of the summarizer unit is depicted in Figure 2(b). When developing the extractive summarizer, we have focused on two discourse aspects: sentence-level semantic connections for information coherence and the influence of word-level semantics on sentence correlations. With these considerations, for a target document $\mathcal{D}$, we have designed the graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ symbolizes the nodes and $\mathcal{E}$ symbolizes the edges that connect these nodes. The set of nodes $\mathcal{V} = \mathcal{V}_w \cup \mathcal{V}_s$, where $\mathcal{V}_w = \{v_{w_1}, v_{w_2}, ..., v_{w_n}\}$ is the set of all the distinct words, $\mathcal{V}_s = \{v_{s_1}, v_{s_2}, ..., v_{s_m}\}$ denotes the set of sentences in $\mathcal{D}$, and $\mathcal{D}$ contains n unique words and m sentences. $\mathcal{E} = \mathcal{E}_{\mathcal{W}\text{-}\mathcal{S}} \cup \mathcal{E}_{\mathcal{S}\text{-}\mathcal{S}}$ is the edge weight matrix where $\mathcal{E}_{\mathcal{W}\text{-}\mathcal{S}}$ represents the edges between word and sentence nodes, and each element $e_{w_i\text{-}s_j}$ in $\mathcal{E}_{\mathcal{W}\text{-}\mathcal{S}}$ is defined in such a way that $e_{w_i\text{-}s_j} \neq 0$ $(i \in \{1...n\}, j \in \{1...m\})$ if the sentence s_j contains the word w_i. $\mathcal{E}_{s\text{-}s}$ symbolizes the edges between sentences in the document. The sentence nodes are initialized with Longformer [CLS] tokens and the word nodes with:

$$w_i = \frac{\sum_{\forall s_j \in \mathcal{V}_s \wedge e_{w_i\text{-}s_j} \neq 0} vec_{w_{i,j}}}{\sum_{\forall s_j \in \mathcal{V}_s} |e_{w_i\text{-}s_j} \neq 0|} \quad (2)$$

where $|e_{w_i\text{-}s_j} \neq 0|$ is the number of occurrences of the word w_i in $\mathcal{D}$ and $vec_{w_{i,j}}$ symbolizes the Longformer word token for word w_i in sentence s_j. Each word-sentence edge $e_{w\text{-}s_{i,j}} \in \mathcal{E}_{\mathcal{W}\text{-}\mathcal{S}}$ is initialized with the corresponding TF-IDF value. Each cross-sentence edge $e_{s_x\text{-}s_y} \in \mathcal{E}_{\mathcal{S}\text{-}\mathcal{S}}$ is initialized with the cosine similarity between Longformer [CLS] tokens of sentences s_x and s_y.

Scientific articles contain a large number of sentences making operations on fully connected sentence node graphs computationally expensive. As a solution, we have discarded the edge connections between sentence nodes with cosine similarity values below a threshold, $\theta = 0.3$, since experimentally, we have discovered that for $\theta \leq 0.3$, the summarization quality of the model is not affected. To find this optimal cut-off value (θ) we conducted experiments using various cosine similarity thresholds between sentence nodes, ranging from 0.20 to 0.60 with intervals of 0.05. To further reduce the computational overhead, the vocabulary size is reduced by replacing words in the document with common synonyms.

Once the graph $\mathcal{G}$ has been constructed and initialized, a graph attention network (GAT) is applied over the word and sentence nodes in an iterative manner to update them. This GAT layer has been designed by following Wang et al. (2020a). Considering h_i as the hidden state representation of either $v_{w_i} \in \mathcal{V}_\mathcal{W}$ or $v_{s_i} \in \mathcal{V}_\mathcal{S}$, where $h_i \in \mathbb{R}^{d_h}$ and $i \in \{1, ..., (n+m)\}$, the GAT layer (incorporating the edge information) is delineated as:

$$\mu_{i,j} = \texttt{LeakyReLU}(\mathcal{W}_a[\mathcal{W}_q h_i; \mathcal{W}_k h_j; e_{i,j}]) \quad (3)$$

$$\alpha_{i,j} = \frac{\exp(\mu_{i,j})}{\sum_{l \in \mathcal{N}_i} \exp(\mu_{i,l})} \quad (4)$$

$$u_i = \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{i,j} \mathcal{W}_v h_j) \quad (5)$$

where, $\mathcal{W}_v$ $\mathcal{W}_k$, $\mathcal{W}_q$, and $\mathcal{W}_a$ are weight matrices that are updated iteratively. $\mathcal{N}_i$ is the set of 1-hop distant neighbour nodes. The attention value between neighbour nodes h_i and h_j is depicted by $\alpha_{i,j}$. For $\mathcal{K}$ attention heads, this GAT layer is designed as:

$$h_i' = ||_{k=1}^{\mathcal{K}} \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{i,j}^k \mathcal{W}^k h_i) \quad (6)$$

Furthermore, a residual connection has been added to prevent gradient vanishing and the final hidden representation, h_i, is:

$$h_i = h_i' + h_i \quad (7)$$

In the first step of model training, the sentence nodes are updated, influenced by their 1-hop distant word nodes, using the aforementioned GAT layer and the position-wise feed-forward network (FFN) (Wang et al., 2020a):

$$\mathcal{U}_{w \to s}^{(1)} = GAT(\mathcal{H}_s^{(0)}, \mathcal{H}_w^{(0)}, \mathcal{H}_w^{(0)}) \quad (8)$$

$$\mathcal{H}_s^{(1)} = FFN(\mathcal{U}_{w \to s}^{(1)} + \mathcal{H}_s^{(0)}) \quad (9)$$

where $\mathcal{H}_w^0 = \mathcal{V}_w$, and $\mathcal{H}_s^0 = \mathcal{V}_s$. In Eq. 8, $\mathcal{H}_s^0$ has been considered as the attention query matrix, and $\mathcal{H}_w^0$ as both the key and value matrices.

Once the sentence nodes are updated using the adjacent word nodes, in the following step the sentence nodes are updated using cross-sentence correlations, followed by a word node update step using the last-modified sentence node representations. Thus, each iteration comprises a sequence of sentence-sentence, sentence-word, word-sentence and cross-sentence edge updates. At the t^{th} iteration, this operation can be stated as:

$$\mathcal{U}_{s\to s}^{(t+1)} = GAT(\mathcal{H}_s^{(t)}, \mathcal{H}_s^{(t)}, \mathcal{H}_s^{(t)}) \quad (10)$$

$$\mathcal{H}_s'^{(t+1)} = FFN(\mathcal{U}_{s\to s}^{(t+1)} + \mathcal{H}_s^{(t)}) \quad (11)$$

$$\mathcal{U}_{s\to w}^{(t+1)} = GAT(\mathcal{H}_w^{(t)}, \mathcal{H}_s'^{(t+1)}, \mathcal{H}_s'^{(t+1)}) \quad (12)$$

$$\mathcal{H}_w^{(t+1)} = FFN(\mathcal{U}_{s\to w}^{(t+1)} + \mathcal{H}_w^{(t)}) \quad (13)$$

$$\mathcal{U}_{w\to s}^{(t+1)} = GAT(\mathcal{H}_s'^{(t+1)}, \mathcal{H}_w^{(t+1)}, \mathcal{H}_w^{(t+1)}) \quad (14)$$

$$\mathcal{H}_s^{(t+1)} = FFN(\mathcal{U}_{w\to s}^{(t+1)} + \mathcal{H}_s'^{(t+1)}) \quad (15)$$

$$\forall e_{s_i - s_j} \in \mathcal{E}_{\mathcal{S}-\mathcal{S}} = \cos\left(\mathcal{H}_{s_i}^{(t+1)}, \mathcal{H}_{s_j}^{(t+1)}\right) \quad (16)$$

The Longformer decoder has been utilized as the abstractive summarizer following the approach used by Yu et al. (2020).

For each segment, once the abstracts of the referenced articles are extractive- and abstractive-summarized, these two summaries are individually concatenated with the segment. These texts are then fed to their corresponding summarizer unit to produce extractive and abstractive summaries of each segment. At this step of the hierarchy, the accumulated citing statememts are also extractive- and abstractive-summarized. In the last hierarchical step, the extractive and abstractive segment summaries are concatenated with the corresponding summary of the citing statements and fed to the corresponding summarizer unit to produce the final extractive and abstractive summaries of the considered article. Both the extractive and abstractive summarizer units use cross-entropy loss functions ($\mathcal{L}_{ext}$ and $\mathcal{L}_{abs}$, accordingly). The model's loss function, $\mathcal{L}$ is defined as:

$$\mathcal{L} = \mathcal{L}_{ext} + \mathcal{L}_{abs} + \mathcal{L}_{seg} \quad (17)$$

5. Experiments

This section first gives a brief description of the hyper-parameter settings and hardware used for the model implementation and then presents the results achieved by the proposed model described in the previous section on the corpus outlined in Section 3.

The experiments have been conducted on a 48GB NVIDIA RTX A6000 GPU with *batch size* = 5 to accommodate the large number of sentences in the scientific documents. For model training with a small batch size, we have followed the approach of Sefid and Giles (2022). Gradients are collected for ten steps and then the parameters are adjusted. The NOAM scheduler is used to regulate the learning rate. Furthermore, to prevent the exploding gradient problem, we have used gradient clipping. The extractive summarizer is initialised with 768-dimensional Longformer embeddings. After that, the extractive summarizer unit uses the GAT (with 8 attention heads) and the following FFN layer to update the graph nodes. After every forward pass, the abstractive and extractive summarizer units' losses are calculated separately. If either unit's validation loss decreases for 5 continuous epochs, its parameter values are stored and its training is paused for the next 10 iterations. We have trained the model for 200 iterations. The FFN hidden layer size is set to 512. For the parallel training of the summarizers, we have followed the approach proposed by Yu et al. (2020). For the segmentation model, apart from the word embedding dimension, we have replicated the hyper-parameter settings used by Xing et al. (2020). This model is fed with 768-dimensional Longformer word vectors. For all of the experiments we have performed 10-fold cross validation and the results reported here are the means of the experimental outcomes.

We have assessed the segmentation performance using F-1 scores. Like Cho et al. (2022), we have experimented with predicting the first sentence and last sentence of each segment and found that when predicting the first sentence of each segment, the model performs better, which supports the claim in Cho et al. (2022). With the joint training of segmentation and summarization, our segmentation model has achieved 86.19 F-1 score on the segmentation task when predicting the first sentences of the segments. We have also noticed that sentences near the segment boundaries are more prone to be included in the summaries.

In order to assess the efficacy of our model for extractive summarization, we undertake the training and evaluation of the following extractive summarization models with our adapted corpus: (1) BERTSumExt (Liu and Lapata, 2019), an exemplar grounded in BERT; (2) HeterSumGraph (Wang et al., 2020a), a heterogeneously structured graph-based technique that accounts for inter-sentence relationships by incorporating supplementary semantic elements; (3) CGSUM (An et al., 2021), a graph-based summarization model that incorporates the information from the source paper plus the referenced articles; and (4) Lodoss (Cho et al., 2022) which performs the segmentation and summarization tasks in parallel regularized by the determinantal point processes regularizer. In the context of abstractive summarization benchmarking, our experimentation encompasses the utilization of the following models: (1) PTGen+Cov (See et al., 2017), founded upon a hybrid pointer generator net-

Models	On Abstracts as Summaries				On T5-Generated Summaries			
	R-1	R-2	R-L	METEOR	R-1	R-2	R-L	METEOR
Extractive								
BERTSumExt (Liu and Lapata, 2019)	45.63	15.99	41.91	34.89	46.01	16.17	42.18	34.97
HeterSumGraph (Wang et al., 2020a)	46.35	16.22	42.64	35.02	46.81	16.29	42.82	35.16
CGSUM (An et al., 2021)	46.98	17.02	44.17	38.26	46.96	16.96	43.85	37.93
GRETEL (Xie et al., 2022)	47.09	17.16	44.26	38.50	47.14	17.08	44.32	38.42
Lodoss (Cho et al., 2022)	47.17	17.22	44.37	38.61	47.29	17.24	44.47	38.66
Proposed Model (Extractive)	**48.39**	**18.18**	**45.18**	**39.13**	**48.43**	**18.21**	**45.19**	**39.11**
Abstractive								
PTGen+Cov (See et al., 2017)	43.99	14.12	38.16	33.51	43.97	14.10	38.18	33.46
BERTSumAbs (Liu and Lapata, 2019)	45.01	15.33	38.96	34.59	45.02	15.36	39.00	34.64
BERT+CopyTransformer (Aksenov et al., 2020)	45.62	15.78	39.93	34.84	45.54	15.81	39.91	34.88
Proposed Model (Abstractive)	**48.12**	**17.96**	**44.91**	**38.85**	**48.04**	**17.99**	**44.71**	**38.82**

Table 1: Results on the proposed corpus. The results consider both the abstracts and the T5-generated summaries incorporating citation statements as the reference summaries. Best results are in bold font.

work designed to facilitate verbatim transcriptions from the source text; (2) BERTSumAbs (Liu and Lapata, 2019), a model rooted in the BERT architecture; and (3) BERT+CopyTransformer (Aksenov et al., 2020), which leverages BERT-windowing techniques to manage textual content exceeding the inherent BERT window limitations. While training these models, to incorporate the background information, we have concatenated the abstracts of the referenced articles and the considered article following An et al. (2021). The citation statements are also concatenated at the end. The same approach is used for HeterSumGraph and CGSUM. To overcome the token intake limitation of the BERTSumEXT and BERTSumAbs, we have added additional positional encoding which is added randomly and fine-tuned in the training phase (An et al., 2021).

The performances for the prior models and our novel proposal are presented in Table 1 using four commonly used metrics. For reference summaries, we have taken into account not only the paper abstracts but also the summaries that we have produced by amalgamating the abstracts with the citing statements via the T5 framework.

HeterSumGraph scrutinizes immediate associations among words and sentences within textual contexts limited to a maximum of 50 sentences. Conversely, our innovative model not only takes into account these immediate cross-sentence correlations but is also adept at handling more extensive text spans, accommodating up to 3500 words.

Over the sentence-word relationships presented in HeterSumGraph, our model provides inter-sentence correlations. These supplementary functionalities, coupled with the enhanced word and sentence features offered by LongFormer, collectively contribute to a notable enhancement in our model's performance.

CGSUM can take up to two-hop reference articles. For the experiment here, it has been restricted to one-hop to comply with our proposed corpus. However, CGSUM considers all the abstracts from the reference article at once, rather than being used segment by segment. Using reference abstracts segment by segment and utilizing an hierarchical summarization approach over segments allows our model to benefit from the background information in the reference articles where it is needed.

However, it is essential to acknowledge that the heightened capabilities of our model necessitates a commensurate increase in computational time and resource allocation.

In terms of performance, our model demonstrates a substantial gain over other models for the extractive summarization task. The extractive summarizer unit, in our model, has achieved 45.18 Rouge-L (R-L) and 39.13 METEOR scores over the "Abstracts as Summaries" which is 0.81 R-L and 0.52 METEOR higher than Lodoss, which is the best performing model among the other extractive summarizers . Over the "T5 Generated Summaries", our model has outperformed Lodoss by 0.72 R-L and 0.45 METEOR scores by attaining 45.19 R-L and 39.11 METEOR scores.

Like the extractive summarizer unit, our abstractive summarizer unit has also outperformed the other abstractive summarizer units by attaining 44.91 R-L and 38.85 METEOR scores over the "Abstracts as Summaries", and 44.71 R-L and 38.82 METEOR scores over the "T5 Generated Summaries". The best performing model among the considered abstractive summarizers, BERT+CopyTransformer, has achieved 39.93 R-L and 34.84 METEOR over the "Abstracts as Summaries", and 39.91 R-L and 34.88 METEOR over the "T5 Generated Summaries".

To perform the ablation study, different units from the proposed model are discarded and then the performances are reported (see Table 2). Experimental results show that if the word-sentence update step is discarded, the model is affected more than by discarding the sentence-sentence update step. This difference corresponds with our knowing that the sentence nodes are still connected via the word nodes, and suggests that removing the word-

Discarded Unit	R-L	METEOR
Sentence-Sentence update†	43.98	38.68
Word-Sentence update†	42.51	37.22
Abstractive summarizer†	43.95	38.47
Extractive summarizer∗	41.63	36.56
Citation network†	42.17	37.16
Citation network∗	41.74	36.89
Segmentation unit†	43.21	38.14
Segmentation unit∗	42.68	37.79
Synonym replacement†	44.07	38.25
Synonym replacement∗	42.94	37.58

Table 2: Ablation Study on the T5 generated summaries: † indicates the extractive summaries and ∗ indicates the abstractive summaries.

sentence update step has a greater information loss. Furthermore, the results show that replacing uncommon words with corresponding common synonyms not only reduces the computational burden, but also improves the performance and justifies the claim by Wang et al. (2020a) which states that articles containing words with higher node degree not only make the summarization task easier for the deep learning models but also improves the performance.

Another observation that we have drawn from the ablation study is that discarding the extractive summarizer affects the abstractive summarizer more than the extractive summarizer is affected when the abstractive summarizer unit is discarded. These performance drops for the summarizer units also indicate the significance of the parallel training of the extractive and abstractive summarizers. Both the extractive and abstractive summarizer units are affected with a performance drop in both cases when the background information provided by the citation graph or the segmentation units are discarded.

The ablation study also shows that providing background information segment-by-segment rather than providing this information as a unit helps the summarizer model attain better performance.

6. Conclusion

In this paper, we have introduced a scientific document summarization model that leverages references within the article to provide background information and reflects the impact of the cited work on the research community through citation statements. We have created a novel corpus based on a citation graph, encompassing abstracts of reference papers and citing statements for 10,000 scientific articles. This work takes the background information from the reference articles segment-by-segment. To our knowledge, this is the first approach to bridge the gap between the two facets of the citation graph in scientific document summarization. Furthermore, this is the first work where the background information has been applied segment-wise. And our experimental results show that these approaches have allowed the model to attain superior performance compared to the other SOTA works[1].

Limitations

We have trained both the extractive and abstractive summarizer units for a large number of epochs. Though to prevent any unit from being over-fitted we have checked the curve of validation loss after every 5 epochs. This is very computationally expensive and demands a longer period of time for model training.

Acknowledgement

This research is partially funded by The Natural Sciences and Engineering Research Council of Canada (NSERC) through a Discovery Grant to R. E. Mercer.

7. Bibliographical References

Dmitrii Aksenov, Julian Moreno Schneider, Peter Bourgonje, Robert Schwarzenberg, Leonhard Hennig, and Georg Rehm. 2020. Abstractive text summarization based on language model conditioning and locality modeling. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6680–6689.

Nouf Ibrahim Altmami and Mohamed El Bachir Menai. 2022. Automatic summarization of scientific articles: A survey. *Journal of King Saud University-Computer and Information Sciences*, 34(4):1011–1028.

Awais Athar. 2011. Sentiment analysis of citations using sentence structure-based features. In *Proceedings of the ACL 2011 Student Session*, pages 81–87.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Shane Bergsma and Dekang Lin. 2006. Bootstrapping path-based pronoun resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting*

[1]Code and data are available at: https://github.com/sudipta90/ScientificArticleSummarizationCitationGraph.git

of the Association for Computational Linguistics, pages 33–40.

Shane Bergsma and Benjamin Van Durme. 2013. Using conceptual class attributes to characterize social media users. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 710–720.

Sangwoo Cho, Kaiqiang Song, Xiaoyang Wang, Fei Liu, and Dong Yu. 2022. Toward unifying text segmentation and long document summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 106–118.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621.

Arman Cohan and Nazli Goharian. 2018. Scientific document summarization via citation contextualization and scientific discourse. *International Journal on Digital Libraries*, 19(2):287–303.

Wafaa S El-Kassas, Cherif R Salama, Ahmed A Rafea, and Hoda K Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, 165:113679.

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

Supriya Gupta, Aakanksha Sharaff, and Naresh Kumar Nagwani. 2022. Biomedical text summarization: a graph-based ranking approach. In *Applied Information Processing Systems*, pages 147–156. Springer.

Kokil Jaidka, Michihiro Yasunaga, Muthu Kumar Chandrasekaran, Dragomir Radev, and Min-Yen Kan. 2019. The CL-SciSumm shared task 2018: Results and key insights. In *CEUR Proceedings*, volume 2132.

Souvik Kundu. 2023. Citation polarity identification from scientific articles using deep learning methods. Master's thesis, The University of Western Ontario.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Lei Li, Yang Xie, Wei Liu, Yinan Liu, Yafei Jiang, Siya Qi, and Xingyuan Li. 2020. CIST@CL-SciSumm 2020, LongSumm 2020: Automatic scientific document summarization. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 225–234.

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.

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. *arXiv preprint arXiv:1907.11692*.

Yixin Liu, Ansong Ni, Linyong Nan, Budhaditya Deb, Chenguang Zhu, Ahmed Hassan Awadallah, and Dragomir Radev. 2022. Leveraging locality in abstractive text summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6081–6093.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411.

Santosh Kumar Mishra, Naveen Saini, Sriparna Saha, and Pushpak Bhattacharyya. 2022. Scientific document summarization in multi-objective clustering framework. *Applied Intelligence*, 52(2):1520–1543.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gúlçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.

Yifu Qiu and Shay B. Cohen. 2022. Abstractive summarization guided by latent hierarchical document structure. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5317.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena,

Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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.

Athar Sefid and C Lee Giles. 2022. SciBERTSUM: Extractive summarization for scientific documents. In *International Workshop on Document Analysis Systems*, pages 688–701.

Sudipta Singha Roy and Robert E. Mercer. 2022. Building a synthetic biomedical research article citation linkage corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022*, pages 5665–5672.

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

Zhengjue Wang, Zhibin Duan, Hao Zhang, Chaojie Wang, Long Tian, Bo Chen, and Mingyuan Zhou. 2020b. Friendly topic assistant for transformer based abstractive summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 485–497.

Max Welling and Thomas N Kipf. 2016. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR 2017)*.

Qianqian Xie, Jimin Huang, Tulika Saha, and Sophia Ananiadou. 2022. GRETEL: Graph contrastive topic enhanced language model for long document extractive summarization. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6259–6269.

Linzi Xing, Brad Hackinen, Giuseppe Carenini, and Francesco Trebbi. 2020. Improving context modeling in neural topic segmentation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 626–636.

Michihiro Yasunaga, Rui Zhang, Kshitijh 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.

Tiezheng Yu, Dan Su, Wenliang Dai, and Pascale Fung. 2020. Dimsum @LaySumm 20. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 303–309.

Haopeng Zhang, Semih Yavuz, Wojciech Kryscinski, Kazuma Hashimoto, and Yingbo Zhou. 2022. Improving the faithfulness of abstractive summarization via entity coverage control. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 528–535.

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.

Ming Zhong, Pengfei Liu, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2019. Searching for effective neural extractive summarization: What works and what's next. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1049–1058.

8. Language Resource References

Chenxin An, Ming Zhong, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2021. Enhancing scientific papers summarization with citation graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12498–12506.

Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander R Fabbri, Irene Li, Dan Friedman, and Dragomir R Radev. 2019. ScisummNet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7386–7393.