applied sciences

MDPI

Article

Enhancing Abstractive Summarization with Extracted Knowledge Graphs and Multi-Source Transformers

Tong Chen [1,*], Xuewei Wang [1], Tianwei Yue [1], Xiaoyu Bai [2], Cindy X. Le [3] and Wenping Wang [1,*]

[1] Carnegie Mellon University, Pittsburgh, PA 15213, USA; xueweiwa@alumni.cmu.edu (X.W.); tyue@alumni.cmu.edu (T.Y.)
[2] Rice University, Houston, TX 77005, USA; xybai521@gmail.com
[3] Columbia University, New York, NY 10027, USA; xl2738@columbia.edu
* Correspondence: tongc2@alumni.cmu.edu (T.C.); wenpingw@alumni.cmu.edu (W.W.)

Abstract: As the popularity of large language models (LLMs) has risen over the course of the last year, led by GPT-3/4 and especially its productization as ChatGPT, we have witnessed the extensive application of LLMs to text summarization. However, LLMs do not intrinsically have the power to verify the correctness of the information they supply and generate. This research introduces a novel approach to abstractive summarization, aiming to address the limitations of LLMs in that they struggle to understand the truth. The proposed method leverages extracted knowledge graph information and structured semantics as a guide for summarization. Building upon BART, one of the state-of-the-art sequence-to-sequence pre-trained LLMs, multi-source transformer modules are developed as an encoder, which are capable of processing textual and graphical inputs. Decoding is performed based on this enriched encoding to enhance the summary quality. The Wiki-Sum dataset, derived from Wikipedia text dumps, is introduced for evaluation purposes. Comparative experiments with baseline models demonstrate the strengths of the proposed approach in generating informative and relevant summaries. We conclude by presenting our insights into utilizing LLMs with graph external information, which will become a powerful aid towards the goal of factually correct and verified LLMs.

Keywords: abstractive summarization; knowledge graph; multi-source transformers; pre-trained language models

Citation: Chen, T.; Wang, X.; Yue, T.; Bai, X.; Le, C.X.; Wang, W. Enhancing Abstractive Summarization with Extracted Knowledge Graphs and Multi-Source Transformers. *Appl. Sci.* **2023**, *13*, 7753. https://doi.org/10.3390/app13137753

Academic Editor: Andrea Prati

Received: 18 May 2023
Revised: 22 June 2023
Accepted: 26 June 2023
Published: 30 June 2023

1. Introduction

Abstractive summarization is one of the most challenging tasks in natural language processing. A model is required that can learn the context of long documents while generating summaries with the key information preserved. Essentially, models completing this task will extract text patterns, match them to well-learned patterns in the decoder, and regenerate a summary with their own language encoder. Multiple sequence-to-sequence models using attention mechanisms have been proposed for abstractive summarization and perform well [1,2].

Since the advent of large language models like BART (Bidirectional and Auto-Regressive Transformers) and BERT (Bidirectional Encoder Representations from Transformers) [3,4], we have seen significantly more progress in the field of text summarization, and with the availability of GPT (Generative Pre-trained Transformers) and its commercialized version, ChatGPT, introduced by OpenAI [5], large language models and their excellent performance in extracting key information from a multitude of text sources have sparked public and academic interest in LLMs.

Although most of the existing state-of-art models are able to generate smooth, informative summaries both on par with human performance and achieving high scores on automatic evaluation metrics like ROUGE [6], they might not understand puns and product

names, which leads to misunderstanding. Furthermore, since summary text is generated based on the possibility of a word's existence in a certain context, they may not be able to consider the factual information and entity correlation in the source document, which leads to fabricated content [7]. We have seen many cases on social media recently where ChatGPT/New Bing has fabricated plausible but fake records based on the user's request. This poses a huge information threat to users who cannot easily verify content truthfulness, which can also be used by hostile groups to spread misinformation and pollute the online literature.

This study aims to utilize information from knowledge graphs to generate more accurate and coherent summaries with large language models like BART. Furthermore, the knowledge graph provides us with a semantic interpretation of the input, which is critical for explainable summarization. Compared to concise and information-dense news articles such as those from the NYT (*New York Times*) Corpus and the CNN/DailyMail dataset, generating abstracts from information-sparse crowdsourcing articles, like those on Wikipedia, is quite challenging. In order to handle such information sparsity in Wikipedia articles, we argue that the internal semantics can be extracted to focus on entities and relations. We chose to use knowledge graphs to build a structural representation to connect relevant subjects between the sentences and paragraphs.

As for more natural but less formalized texts like Wikipedia articles, complex events related to the same entity or subject are spread over several paragraphs. If we examine the center entity (or essentially, the Wikipedia article's topic subject), it would normally be discussed in the majority of paragraphs, potentially with different names. Therefore, if we can capture this latent semantics structure, it will enable our model to focus on what entities the story is based upon. As a result, we can improve both the readability and the information density of our generated abstract. At the same time, this method should also work well for summarizing abstracts of more concise and formatted text like news and gazette articles.

To this end, our main contributions in this work include:

- Wiki-Sum, a dataset that is extracted from the Wikipedia textural dump, then tokenized and extracted into an abstract summary form;
- MultiBART-GAT, a framework for abstractive text summarization incorporating transformers with graph representation augmentation.

Under the common transformer-based encoder–decoder framework, we use the tokenized regular document embeddings as the input of the text encoder, with graph embeddings from a graph attention encoder that takes in the extracted top relations and entities. The relations and entities are extracted externally using an open information extraction system, namely OpenIE. On the other hand, the hidden vector is used regularly as an input to the BART decoder.

We carry out automatic and human evaluations on our Wiki-Sum dataset, along with the CNN/DailyMail dataset, which is a common, standard, easily accessible text summarization dataset that helps us compare model performance. We discovered that our improved approach improves BARTs performance in several metrics on both datasets; this has given us a better understanding of how to utilize transformers to digest external graphical inputs and provided us with possible directions of how to integrate certified facts as semantics into LLMs in text summarization.

2. Related Work

Abstractive Text Summarization. Typically, a neural text summarization agent takes a source document X_i consisting of sentences represented by a list of word tokens, digests them through an encoder to generate a latent representation, and passes the representation to a decoder that outputs the predicted summary Y_i. The goal of model training is to maximize the conditional likelihood of each article–summary pair $< X_i, Y_i >$ in the whole corpus $< X, Y >$. There are several exemplary models on abstractive text summarization [1,8–10], with BART [4], which is itself a revised version of the BERT transformer-based

language model [3] with extra attention layers and revised training schemes, outperforming many previous models in the domain. BART treats abstractive summarization as a translation task, summarizing article text using the same token set of the input article, and has drastically different input and output sizes compared to real translation, where the token sets are different and the output sizes are similar to the input sizes.

Knowledge Graphs and Graph Attention Models. Knowledge graphs, or knowledge bases, are directed multigraphs as a semantic network containing multiple types of entities and relations with the form

$$< Head(v1), Relation(e1), Tail(v2) >,$$

where Head and Tail are entities, usually proper nouns (nodes in graph), and Relations express connections between entities (edges in the graph). Given a graph G = (V, E), we would like to learn a representation of each vertex $v_i \in V$, which is contextualized by attending over the other vertices to which v_i is connected in G.

The graph attention model we use is based on a Graph Attention Network (GAT) [11], which aims to capture the global context of the graph in a more effective manner. Similar models that have been used to extract features from graph semantics serving the goal of text generation include the graph-to-sequence framework [12], Gated Graph Neural Networks (GGNNs) [13], and Graph Convolutional Networks (GCNs) [14]. The most recent attempt was presented in [15], in which graph networks were viewed as guidance signals that use a separate encoder, but with some shared parameters with the document encoder.

Incorporating Semantics in Representation Learning. We have discovered that there is a particular type of representation learning and sequence generation, in which an internal semantics structure is extracted and incorporated from text based on a given input. Such typical graphical structures include knowledge graphs, sentence structures, and dependency parsing trees.

In [16], their approach comprises a set of staged abstract processes. A Factual Statement Extractor extracts concepts from simplified sentences while maintaining grammatical information, then a linguistic analysis is conducted. Based on this, representations are built around verbs, taking subject and objects if they are present. In [17], concept sets are sampled from several large corpora of image/video captions and paired with human-written sentences as expected outputs, forming a textual description from the knowledge graph structure. The model in ref. [18] is based on a pretrained Transformer Language Model, GPT-2, where Subject–Verb–Object text is generated in templates as a new input and a new fake–true story loss is built to train the generator in parallel. Similar designs also appear in models for question answering [19], conversation generation [20], and text generation from structured text [21,22].

Knowledge graphs are a good source of latent semantics that can form a skeleton for text generation, stressing the text/token selection of abstractive summarization. This idea has been proven possible through papers such as [23–27], and will be continually built upon in the future.

3. Problem Statement

3.1. Encoder–Decoder Framework for Summarization

We can formalize the abstractive summarization task as a sequence-to-sequence problem. To deal with this problem, an encoder–decoder framework is widely used. The encoder encodes the input document $X_i = (x_1, x_2, \ldots, x_n)$ into the intermediate semantics representation c, where x_i denotes the ith word in the document and n is the length of the input sequence. Using this representation, the decoder generates the target summary $Y_i = (y_1, \ldots, y_m)$ with a length of m by modeling the conditional probability. y_i denotes the ith word in the summary.

$$p(Y_i|X_i) = \prod_{t=1}^{m} p(y_t|X_i, y < t) \quad (1)$$

3.2. Using Knowledge Graphs to Augment Summarization

Using an input article with word tokens $X = (x_1, x_2, \ldots, x_n)$, we can extract the relevant nodes and relations set $< V, E >$ from the input article. Both the text and the graph input are sent to the encoder to generate a hidden representation, and then transformed in the decoder to form the set of target text.

As for training objectives, we choose to use the maximum likelihood training objective, which minimizes the difference between the generated target text and the source summary $\{X_i, Y_i\} \in D$:

$$L_{mle} = -\frac{1}{N}\sum_{i=1}^{N} \log p(Y_i|X_i;\theta) \tag{2}$$

4. Model Formulation

4.1. Multi-Headed Transformers

We use the multi-headed transformer implemented in BART [4], which was derived from the standard sequence-to-sequence transformers designed in [28]. x denotes the input hidden embedding for each transformer layer.

Each encoder layer:

$$x = GeLU(FC(x + SelfAttn(x))$$
$$x = x + FC(x)$$

Each decoder layer:

$$x = GeLU(FC(x + SelfAttn(x))$$
$$x = x + WeightAttn(CrossAttn(x, encoder\_hidden))$$
$$x = GeLU(FC(x))$$
$$x = x + FC(x)$$

Following BART, the activation function uses GeLUs (Gaussian Error Linear Units); cross-attention over the final hidden layer of the encoder is conducted in the decoder, and no extra fully connected layers are used for word prediction.

4.2. Graph Attention

Built on the extracted knowledge graph, we mark actual sentences between subjects and predicates/verbs and between verbs and objects so that the sentence structure is reflected in the graph. By adding reverse edges and self-loops to enrich the content, we can create a graph for each article.

We use Graph Attention Networks (GATs) and add residual connections between layers, following the method in [11]. Each node v_i is represented by a weighted average of its neighbors:

$$\hat{v}_i = v_i + ||_{n=1}^{N} \sum_{v_j \in N(v_i)} \alpha_{i,j}^{n} \mathbf{W}_{0,n} v_j$$
$$\alpha_{i,j}^{n} = softmax((\mathbf{W}_{1,n} v_i)^T (\mathbf{W}_{2,n} v_j)) \tag{3}$$

Following the recommendation in [24], we use N = 4 in our experiments with two layers of GATs. This graph encoder converts document-level knowledge graph context through converting connections to embeddings for each participating entity and relation, and thus they are sent to the decoder as inputs.

4.3. Encoder–Decoder Network

Both encoders and decoders are based on BARTs transformer architecture, but the decoders are altered so that they are compatible with multiple sources of hidden inputs.

Considering the availability of computational power, we use the base model formulation, in which there are six layers for the encoder and six layers for the decoder (the same as the BART base). Compared to the baseline ASGARD (Abstractive Summarization with Graph Augmentation and Semantic-Driven Reward), which only uses RoBERTa-trained embeddings and tokenizers and an LSTM as an encoder, using BART-based models should achieve a better generalization performance.

We can divide training text input into two parts: the raw tokenized text and the extracted knowledge graph. Tokens are matched to pretrained token embeddings, and knowledge graph triples are linked to pretrained embeddings by TransE [29]. Then, after graph attention transformers preprocess the graph embeddings, they are concatenated with token embedding inputs to be sent to the encoder.

The encoder and decoder share embeddings, as auto-encoder models normally do. During training, the decoder takes the encoded hidden output, both the text from the transformer encoder and the graph embeddings from the graph attention agent, then the decoder conducts multi-source BART decoding. During training, the decoder output is coupled with cross-entropy loss; during inference, the decoder output is multiplied to the shared token embeddings to generate the output distribution, upon which greedy or beam search is applied(As shown in Figure 1).

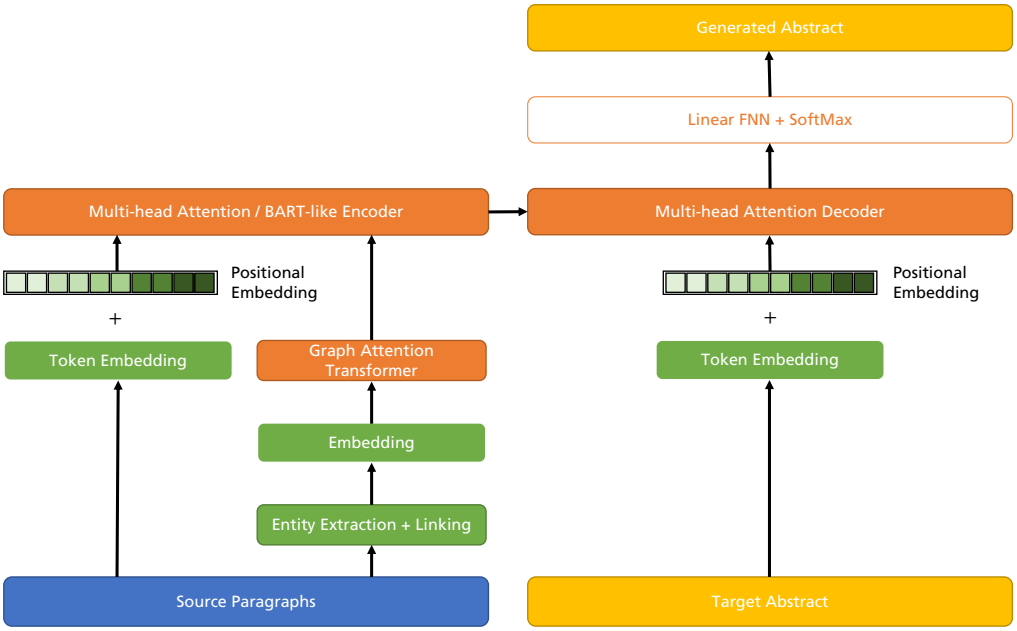

Figure 1. The model framework.

4.4. Initialization, Training, and Loss Function

Considering the cost of training transformers and our current lack of computational resources, we start directly with pre-trained BART base embedding and transformer parameters, then finetune the model given our sub-task definition and input and output structure.

For the loss side, firstly we set up textual-side loss so that the generated abstract can be compared with the ground truth. We use cross-entropy loss between the generated and truth tokens, as described above, which minimizes the difference between the two.

As we have extra information from the knowledge graph, we further add the entity salience objective, which forces the model to predict if entities appear in the abstract. The ground truth should mask the entity with 1 or otherwise 0. Hence, we can introduce the loss function as below.

$$L_{entity} = -\frac{1}{N}\sum_{i=1}^{N} \log p(Y_i|X_i;\theta) \tag{4}$$

5. Dataset: Wiki-Sum

Wiki-Sum is a summarization dataset we propose as part of our idea to promote generating longer, more informative, and more professional abstracts of a passage in comparison to more general, formatted, and commonsensical aspects from news text. We have compiled an automatically tokenized and converted dataset that has been processed by us; however, based on the fact that Wikipedia text does not have predefined summary sentences or highlighted sentences and that the information on Wikipedia can be very diverse, we also created a raw extracted text version without extracted summaries and a processor that picks out sentences in an article.

5.1. Data Collection

Although Wikipedia article texts are in an open-source domain and can be crawled using a crawler, it would be resource intensive to do so; thus, we used a Wikipedia history dump as our source corpus. We downloaded the whole Wikipedia dump at a particular timestamp and selected the 94,000 most-read articles, excluding the home page, lists, and stubs with only one or two paragraphs. To generate this most-read articles list, we retrieved the most-read articles every hour in the month of October 2022, then we mapped these counts to article IDs and reduced the statistics together by the ID. In this way, we will generate a more accurate top articles list and avoid random factors that may cause some articles to be suddenly discovered by readers due to an instantaneous event, like newly announced awards and their affiliated people, breaking news and related locations, launches of products and cultural items, etc.

We selected these articles based on our assumption that these articles are the most viewed. Based on the nature of Wikipedia, which anyone can edit, they should also be the most edited and organized are thus clearly comprehended by most people. Furthermore, since these are the most viewed articles, they should be close to the topics of interest of the general public, in contrast to some articles that only serve particular groups of people, such as topics on maths, pharmacy, animals, merchandise, etc.

We inspected the most likely topics of the selected articles by identifying the key words in the Wikipedia articles, and we observed that our assumption holds and the articles mainly discuss topics that the general public are more interested about. However, there are some articles that are focused on technological topics. This will be exhibited in the dataset analyses later.

Compared to other datasets like CNN/DailyMail, we have found that our Wikipedia-extracted article dataset is much longer, with an average length of 2912 words. Hence, it is innately challenging for a neural model to learn an appropriate representation of the article, considering that LSTMs can only track sequence lengths of hundreds of tokens and transformers can lose focus due to the high repetition of topic words. The reason behind these extremely long articles is that, considering that these are high-traffic pages, the more people that read the article, then the more authors are willing to add timely information and details to that article, ultimately extending the article into thousands of words that might be not be able to be concluded in an abstract.

In order to avoid extremely long documents, which also pushes our computation resources to the limit, and make the dataset computationally feasible, we only selected the first three sentences in each paragraph, minus the first one, to represent the whole input article. For the summary-extracted dataset in our experiments below, to extract a summary from the given text, we used the first three sentences and the title of the head paragraph (or the whole paragraph if the number of sentences was not enough) to conclude the whole article. From our observations, when the Wikipedia article is a narrative, for example, for a biography or history of something, then the first sentence is typically not a summary of the whole paragraph, instead the narrative starts immediately. For other articles, this offers a good snapshot of the paragraph.

We present our dataset statistics in Table 1. We also plotted the article length distribution for the shortened dataset in Figure 2.

Table 1. Dataset statistics regarding the number of articles.

Training set	77,545
Validation set	9697
Test set	9698

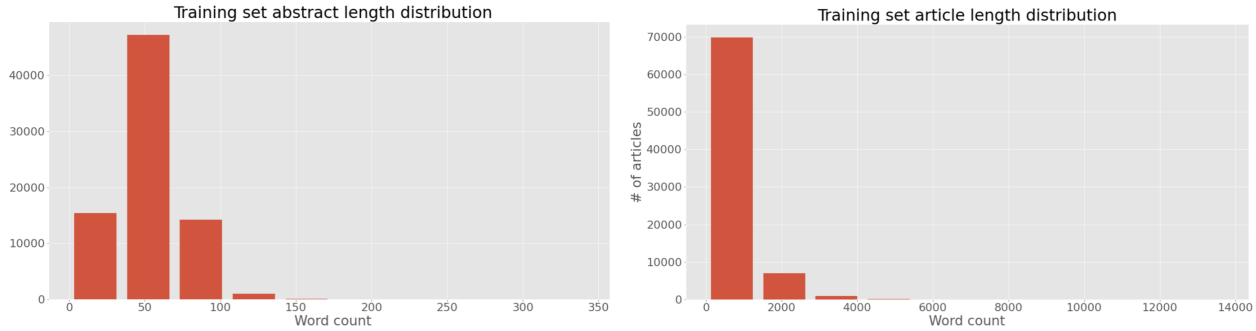

Figure 2. Document length statistics for the training set. (**Left**) length distribution of the summaries; (**right**) length distribution of the articles.

5.2. Dataset Analysis

In order to analyze the content of the Wiki-Sum dataset, we constructed a world cloud for the validation data split in Figure 3. As shown in the figure, the word count for abstracts and articles is relatively consistent. Words like "first", "people", "united states", "film", and "time" are frequent words that occur in the dataset. We also trained a Latent Dirichlet Allocation (LDA) to automatically find mixtures of similar words, thus analyzing the topics or themes in the Wiki-Sum dataset. We discovered eight hidden topics containing three words with the highest scores, which are presented in Table 2.

Table 2. Results of topic modeling of the Wiki-Sum dataset. The categories of the topics were determined by the top three words from the articles belonging to the associated withpics.

Topic	Words
Topic 1 (history)	war, force, army
Topic 2 (literature)	write, century, work
Topic 3 (sports)	season, game, team
Topic 4 (media)	film, release, series
Topic 5 (geography)	city, north, south
Topic 6 (politics)	government, party, united
Topic 7 (education)	university, school, program
Topic 8 (technology)	system, formula, process

We also explored the subjects of these articles by the category of the selected articles and confirmed our assumption that the selected articles mostly fall into the following topics:

- **Locations**: `Canada; Akron, Ohio; Gwangju; Baja California Peninsula`
- **People**: `Lee Se-young; Sandeep Vanga; Harvey Weinstein; Shahid Khan`
- **Concepts**: `Civil engineering; Aryan race; Denial-of-service attack; Student's t-test`
- **Films, TV Series, Literature**: `Doctor Who; The Last of Us; The Exorcist (film); Designated Survivor (season 2)`
- **Organizations**: `Ferrari; Roman Republic; New York Yankees; Princeton University`
- **Events**: `Spanish Civil War; Hurricane Irma; European colonization of the Americas; 2017-18 NHL season`

On the next page are some examples of the training samples in Table 3. We can see that these samples cover topics that are more commonsensical to the human mind, can be more easily extracted, and fit better to existing knowledge graphs like FreeBase.

Table 3. Examples from the Wiki-Sum dataset. We have not incorporated the full text because it would be too long to fit on the page.

Text	Summary
Hoffa was born in Brazil, Indiana, on 14 February 1913, to John and Viola (née Riddle) Hoffa. From an early age, Hoffa was a union activist, and he became an important regional figure with the IBT by their mid-twenties. By 1952, he was the national vice-president of the IBT and was its general president between 1957 and 1971. He secured the first national agreement for teamsters' rates in 1964 with the National Master Freight Agreement. He played a major role in the growth and the development of the union, which eventually became the largest by membership in the United States, with over 2.3 million members at its peak, during their terms as its leader. Hoffa became involved with organized crime from the early years of their Teamsters work, a connection that continued until their disappearance in 1975. He was convicted of jury tampering, attempted bribery, conspiracy, and mail and wire fraud in 1964 in two separate trials. He was imprisoned in 1967 and sentenced to 13 years. In mid-1971, he resigned as president of the union as part of a commutation agreement with US President Richard Nixon and was released later that year, but Hoffa was barred from union activities until 1980. Hoping to regain support and to return to IBT leadership, he unsuccessfully tried to overturn the order. Hoffa disappeared on 30 July 1975. He is believed to have been murdered by the Mafia and was declared legally dead in 1982. Hoffa's legacy continues to stir debate.	
…	James Riddle Hoffa (born 14 February 1913—disappeared 30 July 1975, declared dead 30 July 1982) was an American labor union leader who served as the president of the International Brotherhood of Teamsters (IBT) from 1957 until 1971.
The origins of the company are complex, going back to the early 20th century and the initial enterprises (Horch and the Audiwerke) founded by engineer August Horch; and two other manufacturers (DKW and Wanderer), leading to the foundation of Auto Union in 1932. The modern era of Audi essentially began in the 1960s when Auto Union was acquired by Volkswagen from Daimler-Benz. After relaunching the Audi brand with the 1965 introduction of the Audi F103 series, Volkswagen merged Auto Union with NSU Motorenwerke in 1969, thus creating the present day form of the company. The company name is based on the Latin translation of the surname of the founder, August Horch. "Horch", meaning "listen" in German, becomes "audi" in Latin. The four rings of the Audi logo each represent one of four car companies that banded together to create Audi's predecessor company, Auto Union. Audi's slogan is Vorsprung durch Technik, meaning "Being Ahead through Technology". Audi, along with fellow German marques BMW and Mercedes-Benz, is among the best-selling luxury automobile brands in the world.	
…	Audi AG is a German automobile manufacturer that designs, engineers, produces, markets and distributes luxury vehicles. Audi is a wholly owned subsidiary of the Volkswagen Group and has its roots at Ingolstadt, Bavaria, Germany. Audi-branded vehicles are produced in nine production facilities worldwide.

Table 3. *Cont.*

Text	Summary
The balance of trade forms part of the current account, which includes other transactions such as income from the net international investment position as well as international aid. If the current account is in surplus, the country's net international asset position increases correspondingly. Equally, a deficit decreases the net international asset position. Many countries in early modern Europe adopted a policy of mercantilism, which theorized that a trade surplus was beneficial to a country, among other elements such as colonialism and trade barriers with other countries and their colonies. (Bullionism was an early philosophy supporting mercantilism.) In March 2019, Armenia recorded a trade deficit of US$203.9 million. For the last two decades, the Armenian trade balance has been negative, reaching an all-time high of –33.98 USD million in August 2003. The reason for the trade deficit is that Armenia's foreign trade is limited by its landlocked location and border disputes with Turkey and Azerbaijan, to the west and east, respectively. The situation results in the country's typically reporting large trade deficits.	
…	The balance of trade, commercial balance, or net exports (sometimes symbolized as NX) is the difference between the monetary value of a nation's exports and imports over a certain time period. Sometimes a distinction is made between a balance of trade for goods versus one for services. The balance of trade measures the flow of exports and imports over a given period of time.

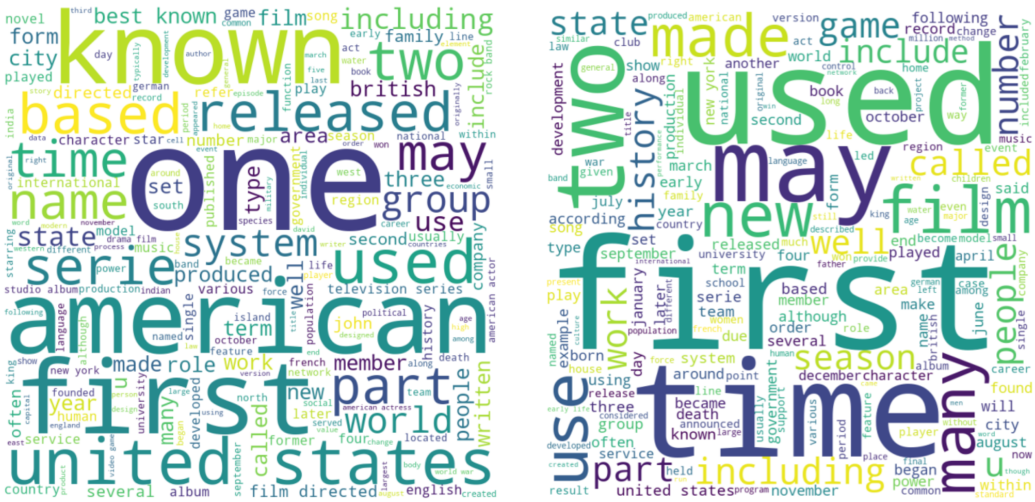

Figure 3. Word cloud plots for articles and abstracts in the validation data split. The left figure is the word cloud for abstracts and the right figure is the word cloud for Wikipedia articles.

5.3. Knowledge Graph Construction

We used OpenIE [30] to extract entity-relation triples with coreference resolution. Since the extracted triples are noisy, we only keep the triples whose subject and object have less than 10 words. We also notice that there are triples with words overlapping, so we filtered out the duplicated triples if the triples only **differ from one argument** and the **arguments have overlaps**, keeping the longest one. The reason for this is that if the triples only differ from one argument (subject or object) and they still have overlaps in that argument, they are likely to be duplicates. As we apply the coreference resolution for the article, we are

able to merge nodes in the graph into their coreferential mention, thus reducing the size of the entire set. As a result, we can generate a graph with subjects and objects as nodes and predicates as relations.

6. Experiments

6.1. Baselines and Comparison

The two main comparable models we selected are ASGARD [27] and BART [4]. The former is the basis of our model implementation, which uses RoBERTa-trained embeddings to initialize word token embedding, Graph Attention to extract graphical features, and LSTMs to encode and decode text with the help of self-attention. We use the default settings of ASGARD, with 256-dimensional encoders and 128-dimensional decoders on the text side and four heads with 72 dimensions on the graph encoder. Considering the lack of cloze reward data sources, we only compared our model and the SEGGRAPH version with no reward training at a learning rate of 1×10^{-3}. The latter, BART, is one of the state-of-the-art pretrained models that is appropriate for many language tasks and uses transformers as core components. We also used the default settings, with a hidden vector size of 1024, and started from the pretrained model. For a fair comparison between BART and our approach, and considering the cost of training, we used the base model with a total of six encoders and six decoders. Furthermore, we used the base training method that does not include a cloze reward or other reinforcement losses, and trained all models for 10 epochs (since training until convergence was too costly).

6.2. Metrics: ROUGE

In the context of text summarization, we used ROUGE as the main metric of text quality comparison because, for one, it is the most widely used automated metric in the field, having been working as a golden standard in the research community for a long time. For another, it is a very efficient way to evaluate how valid a generated summarization is when good methods of human evaluation are lacking.

ROUGE is a metric system for evaluating the automatic summarization of texts, as well as machine translations, which was introduced in [6]. Although ROUGE is 'recall oriented' as its name suggests, its current form considers both precision and recall for the generated target text against its reference source of truth.

ROUGE scores are typically reported as ROUGE-1, ROUGE-2, or ROUGE-L. These metric branches mainly differ by how long a subsequence is compared between the source and target texts; ROUGE-2 works on bi-grams, ROUGE-1 on uni-grams, and ROUGE-L on Longest Common Subsequences (LCSs). To make the scores even more concise, the F1 score is typically used, which is the harmonic mean between precision and recall. This is also the actual value we report.

6.3. Data Augmentation on Our Dataset

We reduced some texts so that they still retain most information that should be included in the summary, while being short enough for transformers to intake and while also not exceeding memory limits. Specifically, we divided the whole article by the original subsections recorded in Wikipedia and selected only the first three sentences in each subsection. This effectively reduced the dataset to a size of 711. Still, we set the maximum length of input tokens to 2048, so that most articles were properly trained while not exploiting too much of our GPU and saving on training time.

6.4. Running Models with Reduced CNN-DailyMail Dataset

Considering that we do not have sufficient access to computational resources, we set up our own workbench with one Nvidia RTX 2080Ti. However, it was still hard to properly train the model in a relatively short amount of time.

To make training faster, we randomly downsampled 10% from all articles, so that the distribution of text was still preserved. After downsampling, we used 30 K articles as the

training dataset. The testing dataset was still the original CNN/DailyMail dataset. We also tested the two baseline models used as references, and also compared them to the reduced CNN/DailyMail dataset. The results are for reference only, since a reduced dataset will certainly constrain the model performance on the whole dataset.

7. Results

7.1. Automatic Evaluation

We conducted an automatic evaluation of the generated summaries using ROUGE scores on the test set beam search outputs of the three models. The results are shown in Tables 4 and 5.

Table 4. Results for the reduced CNN/DailyMail dataset on ASGARD, BART, and MultiBART-GAT compared to the results on the full set. Performances are lower than originally reported values, which is expected because of the smaller amount of training data.

Models	Original			Reduced		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
BART	44.16	21.28	40.90	24.31	10.99	22.91
ASGARD	43.93	20.37	40.48	36.61	14.82	33.73
MultiBART-GAT	35.74	20.03	35.44	27.73	10.06	27.17

Table 5. Results for the Wiki-Sum dataset on ASGARD, BART, and MultiBART-GAT.

Models	Wiki-Sum		
	ROUGE-1	ROUGE-2	ROUGE-L
BART	29.24	8.57	26.62
ASGARD	31.74	13.48	29.02
MultiBART-GAT	30.27	10.02	24.101

7.2. Result Analysis

In the high level comparison, for the CNN/DailyMail dataset, the BART model achieves the best performance on the original dataset and ASGARD outperforms other models in the reduced version. This can be seen from comparing columns 2 and 5 in Table 4. However, when comparing columns 2 and 3 in Table 5 for BART and our proposed model MultiBART-GAT, we can see our modification to BART has a clear positive impact in generating longer, more informative, and more professional abstracts, which is what the Wiki-Sum dataset is targeting. The results are well-aligned with our hypothesis that extracted knowledge graphs help multi-source transformers and other large language models to produce informative and relevant summaries.

Human empirical evaluations also verify that with the guidance of prior knowledge, the readability and topic relevance of the BART-generated summaries have significantly improved, which the original BART model fails to achieve. BART manages to generate paragraphs in a similar style and wording to Wikipedia or news text, but fails to deliver much topical information.

Interestingly, when comparing ASGARD and MultiBart-GAT in columns 1–3 in Table 5, it seems that the LSTM-based encoder used in ASGARD is better than our model. After some analysis, we think that this does not mean that the LSTM-based encoder outperforms BART in general; instead, the poor performance of BART is due to a lack of context introduced from the smaller dataset. This is evident when we compare the results of BART on the original and the reduced dataset. We conclude that the text length is a very important factor for abstractive text summarization models based on LLMs to compare with extractive models.

We also note that the results for all models on the Wiki-Sum dataset are slightly lower on all models compared to both versions of the CNN/DM dataset. Our hypothesis is that there may be a lack of information that Wikipedia authors tend to supply in the bulk of main text and the summary. This is a worthwhile direction of study to improve the dataset.

8. Conclusions

In summary, our research presents a novel approach to abstractive summarization that addresses limitations in current large language models (LLMs). By incorporating knowledge graph information and structured semantics, we enhance the factual correctness. The MultiBART-GAT model, based on BART, serves as an encoder for textual and graphical inputs, improving the summary quality. An evaluation on the Wiki-Sum dataset, derived from Wikipedia, demonstrates its superior performance compared to baseline models in generating informative summaries. As shown in Tables 6 and 7, our research gives insights into leveraging knowledge graphs for more accurate and verified LLM-generated summaries. We also point out potential directions for improvement:

- **Dataset cleaning**. To make the dataset more appropriate, it would be better to conduct data pruning by hand or through mechanical turks to form more concise versions of long Wikipedia articles that preserve the majority of information. Stubs still included in the dataset or special articles without an introduction should also be removed. Furthermore, it is worth mentioning that abstracts for Wikipedia articles describing names and places usually contain information (e.g., full names and birth and death dates) that is exclusively saved in the infobox on the side. We think it would be more appropriate for these facts to be included in the source article in some form, such as in automatically generated sentences.
- **Model design**. On the one hand, other graph encoders like GCN or GGNN should also be considered as concatenated structural inputs. On the other hand, inputs to the encoder can also include extracted entities in the form of emphasized tags. This is also a way to include internal semantics as input enrichment.
- **Training**. We think that there is much room for improvement regarding the speed of model convergence; one way to do this is to use a loss that focuses more on node selection.
- **Future directions**. In the final design of our model, we wish to incorporate a multi-hop generation that controls the story flow coupled with regularly trained language models that have been converted for text generation. This will lead to an LLM that is able to align to verified true storylines and make fewer factual errors.

Table 6. Examples generated from the Wiki-Sum dataset. We use BART and ASGARD samples to show the dataset's quality.

Ground Truth	Summary by BART	Summary by ASGARD
Georgia is a transcontinental country in the Caucasus region of Eurasia. Located at the crossroads of Western Asia and Eastern Europe, it is bounded to the west by the Black Sea, to the north by Russia, to the east by Azerbaijan, and to the south by Armenia and Turkey. The capital and largest city is Tbilisi.	The Roman Empire, also known as the Roman Republic, was a Roman catholic church in the United States. It was founded in 1848 by the Russian Empire, and was the second largest in the world. It is one of the most populous city of the world's most populous state.	Georgia, also known as the Georgia or Georgian, is a feudal region of the Persian peninsula. It is the largest of the world's largest city in the world.
Blues Brothers 2000 is a 1998 American musical comedy film that is a sequel to the 1980 film The Blues Brothers, written and produced by John Landis and Dan Aykroyd. Directed by Landis, the film stars Aykroyd and John Goodman, with cameo appearances by various musicians. The film is dedicated to John Belushi, Cab Calloway, and John Candy, cast members from the original film who had died prior to the sequel's production, as well as Junior Wells, who died one month before it was released.	David is a 2017 American comedy–drama film written and directed by James Walt Disney. The film is based on the novel of the same name by Jimmy Lee. It is the second installment in the "Star Trek" film series.	Y Blues Brothers 2000 is a 1998 American musical film directed by Jim Belushi, starring Joe Morton, and Joliet John Belushi. The film was released on 14 October 2000.

Table 6. *Cont.*

Ground Truth	Summary by BART	Summary by ASGARD
Redlining is the systematic denial of various services or goods by federal government agencies, local governments, or the private sector either directly or through the selective raising of prices. This is often manifested by placing strict criteria on specific services and goods that often disadvantage poor and minority communities.	The U.S. state company is an American multinational company headquartered in New York City, California. The company was founded in the United States, with the company's largest company in the world, and was founded by the United Kingdom and the company of London.	Redlining, also known as the redlining or redlining, is a practice that is used to refer to the practice of the Housing Act of the United States.

Table 7. Examples generated by models.

Ground Truth	Summary by ASGARD	Summary by MultiBART-GAT
Novak Djokovic beat Andy Murray 7-6 4-6 6-0 in Miami Open 2015 final. Djokovic lost their cool after losing the second set to the Brit in Florida. World No 1 Djokovic shouted at their support team next to a scared ball boy. After seeing the replay, the Serbian posted an apology video on Facebook.	Novak Djokovic issued an apology to a ball boy he frightened during the Miami Open final to Andy Murray. The world No 1 grabbed a towel to the ball boy's parents.	Novak Djokovic has posted an apology video on Facebook after losing their cool after losing the Miami Open final to Andy Murray. The Serbian world No 1 lost their cool in the second set of the final against the Brit and shouted at their support team next to a scared ball boy as he was beaten 7-6 4-6 6-0 by Murray in the final of the tournament in Florida on Sunday.
Alfred Taubman, who died on friday, was active in philanthropy and worth an estimated USD 3.1 billion. Amid the suburban boom of the '50s', he realized people would need places to shop: "we could not miss." We were convicted in 2002 of trying to rig auction house commissions; he maintained he was innocent.	The Robert Alfred S. Alfred Taubman died on Friday at their home of a heart attack at a home after a heart at a heart of their heart attack. He donated and manages owns and manages 19 regional shopping centers and owns 19 regional and currently owns 19 and owns and a manages 19 years ago.	Alfred Taubman, a billionaire who was convicted in 2002 of trying to rig auction house commissions, has died at the age of 90. Taubmann, who was born in New York City, was the co-founder of We Auctions, one of the nation's largest auction houses. He died Friday at their home in the Bronx, New York, after a long battle with cancer, according to a statement from their family.

Author Contributions: Conceptualization, T.C. and X.W.; methodology, T.C.; software, X.W. and T.Y.; formal analysis, T.Y., X.B.; data curation, C.X.L.; writing—original draft preparation, T.C., X.W.; writing—review and editing, W.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The study did not report any data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. See, A.; Liu, P.J.; Manning, C.D. Get to the point: Summarization with pointer-generator networks. *arXiv* **2017**, arXiv:1704.04368.
2. Nallapati, R.; Zhou, B.; dos Santos, C.; Gulçehre, Ç.; Xiang, B. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In Proceedings of the Conference on Computational Natural Language Learning, Berlin, Germany, 7–12 August 2016.
3. Kenton, J.D.M.W.C.; Toutanova, L.K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the NAACL-HLT, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
4. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv* **2019**, arXiv:1910.13461.
5. OpenAI. GPT-4 Technical Report. 2023. Available online: http://xxx.lanl.gov/abs/2303.08774 (accessed on 24 June 2023).
6. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*; Association for Computational Linguistics: Barcelona, Spain, 2004; pp. 74–81.
7. Cao, Z.; Wei, F.; Li, W.; Li, S. Faithful to the original: Fact-aware neural abstractive summarization. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, New Orleans, LA, USA, 2–7 February 2018; pp. 4784–4791.

8. Paulus, R.; Xiong, C.; Socher, R. A Deep Reinforced Model for Abstractive Summarization. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
9. Gehrmann, S.; Deng, Y.; Rush, A.M. Bottom-up abstractive summarization. *arXiv* **2018**, arXiv:1808.10792.
10. Dong, L.; Yang, N.; Wang, W.; Wei, F.; Liu, X.; Wang, Y.; Gao, J.; Zhou, M.; Hon, H.W. Unified language model pre-training for natural language understanding and generation. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 13063–13075.
11. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903.
12. Song, L.; Zhang, Y.; Wang, Z.; Gildea, D. A Graph-to-Sequence Model for AMR-to-Text Generation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1: Long Papers, pp. 1616–1626.
13. Beck, D.; Haffari, G.; Cohn, T. Graph-to-Sequence Learning using Gated Graph Neural Networks. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1: Long Papers, pp. 273–283.
14. Damonte, M.; Cohen, S.B. Structural Neural Encoders for AMR-to-text Generation. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1 (Long and Short Papers), pp. 3649–3658.
15. Dou, Z.Y.; Liu, P.; Hayashi, H.; Jiang, Z.; Neubig, G. GSum: A General Framework for Guided Neural Abstractive Summarization. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; pp. 4830–4842.
16. Vodolazova, T.; Lloret, E. The Impact of Rule-Based Text Generation on the Quality of Abstractive Summaries. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), Varna, Bulgaria, 2–4 September 2019; pp. 1275–1284.
17. Lin, B.Y.; Shen, M.; Xing, Y.; Zhou, P.; Ren, X. CommonGen: A constrained text generation dataset towards generative commonsense reasoning. *arXiv* **2019**, arXiv:1911.03705.
18. Guan, J.; Huang, F.; Zhao, Z.; Zhu, X.; Huang, M. A knowledge-enhanced pretraining model for commonsense story generation. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 93–108. [CrossRef]
19. Dhingra, B.; Zaheer, M.; Balachandran, V.; Neubig, G.; Salakhutdinov, R.; Cohen, W.W. Differentiable reasoning over a virtual knowledge base. *arXiv* **2020**, arXiv:2002.10640.
20. Zhou, H.; Young, T.; Huang, M.; Zhao, H.; Xu, J.; Zhu, X. Commonsense Knowledge Aware Conversation Generation with Graph Attention. In Proceedings of the IJCAI, Stockholm, Sweden, 13–19 July 2018; pp. 4623–4629.
21. Hajdik, V.; Buys, J.; Goodman, M.W.; Bender, E.M. Neural Text Generation from Rich Semantic Representations. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1 (Long and Short Papers), pp. 2259–2266.
22. Marcheggiani, D.; Perez-Beltrachini, L. Deep Graph Convolutional Encoders for Structured Data to Text Generation. In Proceedings of the 11th International Conference on Natural Language Generation, Tilburg, The Netherlands, 5–8 November 2018; pp. 1–9.
23. Wu, Z.; Koncel-Kedziorski, R.; Ostendorf, M.; Hajishirzi, H. Extracting Summary Knowledge Graphs from Long Documents. *arXiv* **2020**, arXiv:2009.09162.
24. Hou, S.; Lu, R. Knowledge-guided unsupervised rhetorical parsing for text summarization. *Inf. Syst.* **2020**, *94*, 101615. [CrossRef]
25. Zhu, C.; Hinthorn, W.; Xu, R.; Zeng, Q.; Zeng, M.; Huang, X.; Jiang, M. Boosting factual correctness of abstractive summarization with knowledge graph. *arXiv* **2020**, arXiv:2003.08612.
26. Gunel, B.; Zhu, C.; Zeng, M.; Huang, X. Mind The Facts: Knowledge-Boosted Coherent Abstractive Text Summarization. *arXiv* **2020**, arXiv:2006.15435.
27. Huang, L.; Wu, L.; Wang, L. Knowledge Graph-Augmented Abstractive Summarization with Semantic-Driven Cloze Reward. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020.
28. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
29. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating embeddings for modeling multi-relational data. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 2787–2795.
30. Angeli, G.; Premkumar, M.J.J.; Manning, C.D. Leveraging linguistic structure for open domain information extraction. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 26–31 July 2015; Volume 1: Long Papers, pp. 344–354.