Transformer-based Contrastive Summarization Tool on Amazon Product Reviews

Yuyin Lang
Department of Informatics
Technical University of Munich
yuyin.lang@tum.de

Abstract

Summarization is used in many applications in life. Question-answering bots use summarization to understand relevant texts they gathered quickly, and investment banking firms need summarization to get track of numerous market reports and news every day, to list a few. In this work, we developed a tool to help Amazon consumers understand the pros and cons of the products quickly so they can make the right purchase decision. This tool takes the raw reviews as input, separates the sentences, and identifies those praising the product (positive sentences) and those criticizing it (negative sentences). Then, it generates contrastive summarizations for the product: One for positive and negative sentences. Finally, it uses SUM-QE to evaluate the quality of the generated summarizations, which gives the user insight into how good the summarization is. The complete code of this work can be found here[1].

1 Introduction

In the last decade, online shopping has increasingly become essential in our daily lives. It is much more convenient than traditional shopping in the mall – only by clicking the mouse at home will you find the packet in front of the door in the next few days. This online shopping demand even arrived at another top during the pandemic, when access to onsite shopping was restricted. When shopping online, for example, on Amazon, product reviews are essential. They provide insights from buyers' views and are thus quite good references for other buyers. However, reading numerous reviews for different products is never easy, especially when the buyer must remember competing products' strengths and drawbacks.

To solve this challenge, we designed a Transformer-based tool based on models which can analyze product reviews automatically and generate summarization for both positive and negative aspects of a product to help consumers make proper decisions. The Transformer model (Vaswani et al., 2017) became a sensation as soon as it was proposed. Based on it, BERT (Devlin et al., 2018) even became the SOTA model in various tasks. In this work, we implemented Transformer-based models in three ways:

- We used BERT as a sentiment analysis model to classify Amazon raw product reviews into positive and negative clusters.
- With the help of PEGASUS (Zhang et al., 2020), a Transformer-based abstractive summarization model, we created contrastive summarizations for each product.
- We used BERT-based SUM-QE (Xenouleas et al., 2019) to evaluate the quality of the generated summarizations.

We introduce some related work in §2. Next, we elaborate on the dataset preparation process (§3) and generate summarization (§4). In §5, we evaluate the quality of the summarization and do some analysis on it. Then, we conclude this work (§6). Finally, we raised some future work that can be done (§7).

2 Related work

In general, summarization techniques are divided into extractive summarization, which uses some of the sentences in the original text as summarization, and abstractive summarization, which paraphrases the critical idea of the text in other words. There are many summarization model examples. Nallapati et al. (2017) raised an RNN-based sequence model for extractive summarization of documents and showed that it achieves performance better than or comparable to state-of-the-art. Then, after the appearance of BERT, we have seen its significant application in extractive summarization (Liu, 2019).

[1] https://github.com/yuyinlang

In the meantime, BERT is also widely used for abstractive summarization. Liu and Lapata (2019) proposed a new fine-tuning schedule that adopts different optimizers for the encoder and the decoder as a means of alleviating the mismatch between the two, and Zhang et al. (2020) achieved state-of-the-art performance on 12 downstream datasets measured by ROUGE scores with a pre-training large Transformer-based encoder-decoder model on massive text corpora.

This work focuses on a specific kind of summarization: Contrastive summarization. The goal is to compare two opposing groups of texts and generate a summarization for each group, in which the difference between the groups can be seen. A promising pipeline can be found in (Campr and JEŽEK, 2015), where they divide contrastive summarization into three steps: Processing, semantic processing, and summary creation. Different methods were summarized in the survey by Moussa et al. (2018). To add, expert-guided contrastive opinion summarization (Guo et al., 2015) took into account expert opinions and achieved better performance. Our work differs from those in using Transformer-based models to enhance model performance.

There are many datasets available for contrastive summarization. The El Capitan corpus collected by Ibeke et al. (2016) contains 10,349 customer reviews of OS X El Capitan and captures the sentiment and topic information at both the review (document) and sentence levels. On the other hand, the Amazon product review dataset (Jindal and Liu, 2008) contains far more raw reviews from various products, including over 10,000 tagged review sentences. Thus, it is more suitable for this work. Besides, some Tweet datasets are also available for contrastive summarization. For example, Indyref Tweet Dataset (Brigadir et al., 2015) collects over 1 million Tweets concerning Scottish Independence. However, this dataset only saves the Tweet and user ID, making it challenging to classify Tweets supporting and opposing Scottish Independence.

Apart from the short datasets mentioned above, some long datasets are also available for contrastive summarization. Bitterlemons corpus (Lin et al., 2006) collects 594 articles concerning the Israel Palestine conflict. This dataset contains not only article texts but also annotates their topics and viewpoints and even makes a short summary for each piece. Another example is the COCOTRIP dataset (Iso et al., 2021), which contains 100 documents of trip reviews. Although these two datasets are well-designed for contrastive summarization, the number of texts is relatively tiny. Therefore, we did not use them in the task.

ROUGE (Lin, 2004) is a widely used evaluation measure for summarization tasks in the summarization evaluation practice. It compares the generated summarizations and expert-written golden annotated summaries. However, we have to switch to other measures when the golden annotated summaries are unavailable. SUM-QE (Xenouleas et al., 2019) provides five different aspects to measure summarization quality and does not need human-generated summaries. Also, it is proved that this evaluation is highly correlated with human assessment, making it quite suitable for this work.

3 Dataset Preparation

This section introduces how we collect, clean, and prepare the dataset for summarization. After the preparation step, one product's positive and negative sentences will be saved separately and fed into the summarization model in the next step.

3.1 Data source

This work uses two similar datasets: Amazon-tagged product reviews and Amazon raw product reviews. The former dataset contains approximately 638 reviews, among which each sentence is annotated with sentiment; the latter contains over 320k reviews, and no annotation is given. Both datasets were first collected by Jindal and Liu (2008) and can be found here[2]. Note that one has to ask for permission to access the latter dataset.

3.2 Sentiment analysis

In this step, we used Amazon tagged product reviews to train a BERT-based sentiment classification model. Here, we split the reviews into sentences and deleted meaningless ones to obtain a sentence set with approximately 10k sentences. The ground truth of the sentence label is one of the following three kinds:

- **Positive:** Sentences that are annotated with a "+" sign. We do not distinguish sentences that are only slightly positive or highly positive.
- **Negative:** Sentences that are annotated with a "-" sign. Same as positive ones, we do not care about the negative extent.

[2] https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#datasets

- **Neutral:** Sentences that are neither annotated with a "+" nor a "-" sign. These sentences usually appear as a connection of sentences in the storytelling and cannot be ignored.

During the training, we tried out the "BERT-large-uncased" model from Huggingface with different batch sizes (see Table 1). In evaluating the model, we are interested in the accuracy of the model, i. e. how many positive and negative sentences are correctly classified. Here, we introduced strict scores and relaxed scores. For the strict score, only sentences that are predicted correctly can be regarded as a correct match. However, for the relaxed score, one prediction is considered correct as long as it is not classified as the opposite sentiment, which means the classification of sentences with neutral ground truth is always regarded as correct. This idea is simple: It does not matter if the positive sentence group (or negative sentence group) contains some neutral sentences, as they are likely not to influence the summarization output. Therefore, achieving a high relaxed score in this task is enough.

According to table 1, we do not see a model that beats all the other ones. However, we notice that the model achieves nearly the best performance in epoch 3 with a batch size equal to 8. Although the model with batch size equal to 8 wins this model in epoch four by 0.001 for positive sentences, its performance for negative sentences is far worse than the model in epoch 3. Therefore, we chose this model in epoch 3 to continue the next steps.

	batch size = 8		batch size = 16	
epoch	**pos**	**neg**	**pos**	**neg**
1	0.978	0.945	0.976	0.937
2	0.983	0.945	0.978	**0.962**
3	0.983	**0.966**	0.982	0.950
4	**0.984**	0.945	**0.983**	0.954
5	0.978	0.958	**0.983**	0.950

Table 1: BERT-large training output (relaxed score)

3.3 Raw reviews sentiment prediction

With the model trained in Section 3.2, we could classify sentences in the Amazon raw product review dataset as positive and negative. Noting that positive sentences may appear even in 1-star rating reviews and negative ones may appear in 5-star rating reviews, we decided to look at the sentence level and discard the ratings, which means the rating does not play a role in this task as a proxy of sentiment prediction.

As stated above, there are pretty many reviews in the dataset, but we do not need all of them. Hence, we only chose products with between 100 and 200 reviews in this work. Then, we split them into sentences and predicted the sentiment. Finally, we only kept the sentences with a confidence score of higher than 0.95 and saved them to txt files. Each product has two files: Positive and negative.

4 Summarization

In this step, we intended to use an abstractive summarization model to extract the most critical information in the reviews. The extractive summarization is not suitable for this task because it only repeats some sentences it saw, which is not much different from reading reviews directly on Amazon. On the contrary, an abstractive summarization model can understand what the positive and negative reviews are about and thus is more helpful for consumers.

We use the PEGASUS model (Zhang et al., 2020) in this work for summarization. The base architecture of PEGASUS is a standard Transformer encoder-decoder. Like BERT, it uses GSG and MLM to train the Transformer model. But in PEGASUS, GSG and MLM can be applied simultaneously. For example, for the input with three sentences, we can mask the mid-sentence as [MASK1] for the GSG task and mask some tokens randomly for the other two sentences with [MASK2] input in Transformer Encoder simultaneously.

The PEGASUS model has been trained on different corpora, resulting in multiple variants. In this work, we tested five PEGASUS variants with the sentences from Section 3 (“google/pegasus-xsum”, “google/pegasus-large”, “google/pegasus-cnn_dailymail”, “google/pegasus-multi_news”, “google/bigbird-pegasus-large-arxiv”. Examples of summarizations can be found in Table 2.

One can tell the difference between PEGASUS variants from their names: They are pretrained on different datasets. As is shown in Table 2, their performances vary pretty much. The summarization for "google/pegasus-xsum" is relatively brief, and even worse, it creates a positive summary for negative sentences; for "google/pegasus-cnn dailymail," all of the sentences are taken from reviews, which causes it to behave like an extractive sum-

model name	summarization (pos)	summarization (neg)
google/pegasus-xsum	zen vision m is the best mp3 player i have ever owned.	I’ve been using the zen:m for a few weeks and it’s been pretty good.
google/pegasus-large	also the music quality is awesome, i havent heard a single crack in th music, but for better quality you should get new earphones the ones the give are not that great sony has very good ones. a powerful 30 gb can hold what you need...pay a little extra than you would for its "cousins," you will be very pleased! a powerful 30 gb can hold what you need...pay a little extra than you would for its "cousins," you will be very pleased!	but anyway also the video quality blows the ipod video out of the water, it actually feels like you are in the movie theater (kind of). you need a little adapter, its a little bit a pain. unless you’re lucky, all of that needs to be done via the internet....and you will need an extra sync adapter...probably two (mine are on backorder - which has been another problem).
google/pegasus-cnn_dailymail	creative vision:m 30gb mp3 and video player is not all heavy.<n>30gb hard-drive can really be used as a flash drive!!!	Reviewer: "The only downside i found was the lack of a charger in the box"<n>"The only thing iḿ not as happy with is the control pad"<n>Reviewer: "The player is not as fast as the ipod(s)"
google/pegasus-multi_news	– Apple’s latest iPod Touch is getting good reviews from reviewers, but not everyone is a huge fan: “It’s a bit small and not as good as I would have wanted it to be, but it’s still a very good player,” says one. Another says it’s “the best player I’ve ever owned,” but a third says it’s “the worst player I’ve ever owned.” Click here for more.	– If you’ve been dying to get your hands on an iPod Touch, the Consumerist has just the thing for you: Some of the best-selling iPod Touches of all time: Best-selling iPod Touch: 50 million sold Best-selling iPod Touch: 25 million sold Best-selling iPod Touch: 16 million sold Best-selling iPod Touch: 10 million sold Best-selling...
google/bigbird-pegasus-large-arxiv	SALVAGEDATA SALVAGEDATA SALVAGEDATA SALVAGEDATA SALVAGEDATA...	SALVAGEDATA SALVAGEDATA SALVAGEDATA SALVAGEDATA SALVAGEDATA...

Table 2: Summarization examples of PEGASUS

measure	average	std	max	min
pos_Q1	0.3104	0.1200	0.6497	0.0380
pos_Q2	**0.3140**	**0.1194**	0.6512	0.0408
pos_Q3	0.3112	0.1195	0.6518	0.0439
pos_Q4	0.1868	0.1648	0.6544	-0.2005
pos_Q5	0.1924	0.1626	**0.6552**	**-0.2006**
neg_Q1	0.4350	0.1506	0.7846	-0.0397
neg_Q2	**0.4361**	**0.1500**	0.7886	-0.0325
neg_Q3	0.4334	0.1509	0.7862	-0.0439
neg_Q4	0.3552	0.2258	**0.8397**	**-0.2666**
neg_Q5	0.3578	0.2258	0.8377	-0.2525

Table 3: Quality measurement result of SUM-QE

mary model; and things are even worse for both "google/pegasus-multi news" and "google/bigbird-pegasus-large-arxiv": They generate nearly unreadable texts and almost do not convey any useful information. Compared to them, "google/pegasus-large" performed relatively well concerning readability and the proper length. Hence, we used this model to generate summarizations for all the products we collected.

5 Evaluation

In this section, we use SUM-QE (Xenouleas et al., 2019) to evaluate the quality of generated summarizations. SUM-QE stands out for not requiring human-annotated summaries as golden summaries, and it scores a summary in 5 aspects:

- **Q1 -** Grammaticality: The summary should have no datelines, system-internal formatting, capitalization errors or obviously ungrammatical sentences (e.g., fragments, missing components) that make the text difficult to read.
- **Q2 -** Non redundancy: There should be no unnecessary repetition in the summary.
- **Q3 -** Referential Clarity: It should be easy to identify who or what the pronouns and noun phrases in the summary are referring to.
- **Q4 -** Focus: The summary should have a focus; sentences should only contain information that is related to the rest of the summary.
- **Q5 -** Structure & Coherence: The summary should be well-structured and well-organized. The summary should not just be a heap of related information, but should build from sentence to sentence to a coherent body of information about a topic.

After inputting the product reviews to SUM-QE, we obtained the SUM-QE scores with all five quality measures for both positive and negative reviews. We calculated the mean, standard deviation, highest score and lowest score for each quality measure

quality	example
good	as good as ever upgraded from the previous ipod and i found this one to be noticeable better than the first. not only is the music sound quality absolutely amazing, but the new details apple has added make it even more worthwhile. i’ve successfully connected my ipod to my home stereo and the sound quality is equal to a cd.
fair	the sound however is really good, sony knows how to split the stereo up to very good effect, but overall i sold this because i couldn’t listen to what i wanted to. great player this player sounds great and holds a lot of songs. an adequate mini disc recorder the sony mz-ne 410 is a fair to good replacement for my older i have put on and taken off quite a bit of music on this device and the battery consumption has not even moved a bit.
bad	awesome cycling tool the edge 305 provides a wealth of knowledge, is easy to install, as well as use. great product my cycling monitoring with the edge 305 has been easy and comprehensive, it is a great product. garmin edge 305hr+ bicycle monitor and gps (heart rate monitor and cadence) i am very happy with my purchase, the garmin 305 is accurate and reliable.

Table 4: Quality measurement examples (Q4, positive)

over all the products. The result can be seen in Table 3. Here, average stands for the average score over all the products; std stands for standard deviation, max for the highest score, min for lowest score.

SUM-QE uses a pre-trained BERT model adding just a task-specific layer and fine-tuning the entire model on the task of predicting linguistic quality scores manually assigned to summaries. According to the paper’s evaluation, this model achieves very high correlations with human ratings, showing the ability of BERT to model linguistic qualities that relate to text content and form. Thus, it is quite a good fit for our task.

After inputting the product reviews to SUM-QE, we obtained the SUM-QE scores with all five quality measures for both positive and negative reviews. We calculated the mean, standard deviation, highest, and lowest score for each quality measure over all the products. The result can be seen in Table 3. Here, the average stands for the average score over all the products; std stands for standard deviation, max for the highest score, and min for the lowest score. It should be noted that a higher score represents better performance, and the range of the scores is not 0 to 1.

However, the score alone does not speak loud. It can be seen from Table 3 that every quality measure has a different average and std score, making it complicated only to compare numbers between quality measures. Whether a score shows good or bad performance should depend on the respective average and std score. So that this summarization tool gains more practical meaning, we used "good", "fair", or "bad" to describe the quality, dependent on the quality score.

First, we chose summaries whose quality score is close to the maximum, mean, and lowest scores. Here, we decided on Q4 for the comparison because it is the best measure for humans to judge whether a summary is good. The summaries can be found in Table 4.

To recap, Q4 measures focus. As the table suggests, the good example is quite fluent to read, stating this product is much better than the former one and listing the pros. However, the fair example starts by complimenting the product’s superior sound quality but ends up talking about the battery. Finally, multiple topics, such as usability, good monitor, and reliability, can be seen in the bad example.

It is not enough to see the difference between the three sentences as we do not have a specific boundary yet. To do so, we need the help of std. It is intuitive to set the edge for fair sentences as [average – std, average + std]; however, it is too fast to decide without comparing summaries. In Table 5, we showed summaries with different Q4 scores that differ by 0.5std.

In the six summaries shown in Table 5, we can see the quality difference. For the summary with the quality score of average-1.5std and average-std, many topics are covered, and the topics are not entirely relevant to each other; for the summary with a quality score at average-0.5std and average+0.5std, two topics are mentioned; and for the two summaries with highest scores, the topic is quite clear. The summaries with scores between average-0.5std and average+0.5std were therefore designated as fair, those with scores above this range as good, and those below this range as bad. The boundaries can be found in Table 6.

6 Conclusion

This paper introduces the development process of the Amazon product review summarization tool. We first used the open Amazon tagged dataset to train a sentiment analysis BERT model, and it

score	example
average-1.5std	**lousy customer service**, **buggy software** (i still can't download the updates!) every time i try to update it either freezes as soon as it completes or before it even starts. bad tech support this is a good package if you are an experienced computer user who can solve complex problems without tech support. **terrible support** - unless you love their "web" support as other reviewers have stated there is problem importing data from quicken and tc (which intuit is denying). the updates aren't availabe for downloading, to print your forms, you have to download the fonts, to talk to customer service/support help they put you on hold for over a hour and not helpful at all.
average-std	**horrible product and customer service** - stay away if you like being stuck on a technical support line for 30 minutes in the year 2007, please, by all means, use linksys. doesn't function and is backed by unsupportive service we started out with d-link initially because back then it was the cheapest and easiest to access, but recently our adapter had technical problems and then physicall busted. **weak signal strength** i am using the linksys wireless adapter to connect to the wifi networks in cofee shops and the public libraries. the problem is that the software is **not capable of dealing with the router's security**, so you don't get assigned an address – and it doesn't work when you assign yourself one.
average-0.5std	the creative media source software that comes with this product also locks up when an error occurs during file transfer and i have to kill the process to shut it down. **the creative software is not too good** - i did some searching and found some recommendations for notmad explorer - which does truly rock as accessory software for the nomad. **headphone jack problems** the headphone jack on my zen xtra stopped working day 40 (1st day on a 7-day cruise, no less!) breaks easily and has bad warrenty the warrenty on this device is only 3 months and it breaks quite quickly.
average+0.5std	**too much static** and **poor battery quality** this monitor started out fine, but after about 3 months we started getting a ton of static over the monitor making it sometimes difficult to sleep with it on. however, after about 6 months, it started to static whenever i touch the volume control/dial. i, too, have had problems with the dial creating major static to the point that i can't hardly find a location on the dial that will allow us to hear our son without loud crackles of static interference! however, after only 8 months of use, we experienced a loud, crackly static problem every time we adjusted the volume button.
average+std	this is all complicated by the fact that the **supplied documentation is zero help** for this kind of configuration (and questionable for the "normal" one) and, as is often the case with linksys fringe products, there are zero online support resources as in no knowledge base entries for the product, nothing. also frustratingly missing was any description of what network ports the unit uses so that a firewall can be configured properly (again, using internet connection sharing).
average+1.5std	the only real problems i've found, and they are minor, is the **autofocus being a bit slow or inaccurate** (but not terrible) in moving object situations and the fact that when you're finished with macro mode, you need to switch to manual focus.

Table 5: Quality measurement examples with different stds (Q4, negative)

measure	bad-fair	fair-good
pos_Q1	0.2504	0.3704
pos_Q2	0.2543	0.3737
pos_Q3	0.2515	0.3710
pos_Q4	0.1044	0.2692
pos_Q5	0.1111	0.2737
neg_Q1	0.3597	0.5103
neg_Q2	0.3611	0.5111
neg_Q3	0.3580	0.5089
neg_Q4	0.2423	0.4681
neg_Q5	0.2449	0.4707

Table 6: Boundaries of quality measures

achieved the relaxed score of 0.983 and 0.966 for positive and negative sentences on the test set, respectively. Then, we used this model on the larger Amazon raw product review dataset to classify review sentences as positive and negative. Next, we put these sentences into the PEGASUS model to generate abstractive summarization for each product. With the SUM-QE model, we obtained the quality scores of the summarizations in five different aspects and analyzed the result. According to the analysis, we separated the summarization into three different classes: good, fair, and bad, and we further defined the boundaries between classes as [average – std, average + std]. We believe this is important when consumers read the summaries and make shopping decisions.

7 Future work

Possible future work can improve this tool by using Amazon API to obtain product reviews automatically. So far, one needs to enter the product reviews manually to generate the summarization. There might be hundreds of reviews for some popular products, making the Copy-Paste quite tiring. With a good API, one may only have to enter the product ID to see the result.

Another direction of future work can be improving the summarization model. So far, the model only takes English as input, which limits the tool to non-English speaking users. Training a multi-language abstractive summarization can solve this problem.

Finally, the size of the whole model is quite extensive as it uses three Transformer-based models in a row. Especially for the SUM-QE part, the size for one quality measure is 1.2G, making the whole model as big as 6G for all five measures. Such a big size makes it not so practical if the user wants to download it to the local area for future use. Hence, future work on decreasing the model size would be promising.

Acknowledgment

I want to thank Mr. Tobias Eder for supervising me in this guided research project by making some key recommendations. I would also like to thank Prof. Bing Liu from UIC for providing me with the large Amazon raw product review dataset.

References

Igor Brigadir, Derek Greene, and Pádraig Cunningham. 2015. Analyzing discourse communities with distributional semantic models. In *Proceedings of the ACM Web Science Conference*, pages 1–10.

Michal Campr and KAREL JEŽEK. 2015. Contrastive summarization: Comparing opinions of czech senators. *Journal of Theoretical & Applied Information Technology*, 77(1).

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

Jinlong Guo, Yujie Lu, Tatsunori Mori, and Catherine Blake. 2015. Expert-guided contrastive opinion summarization for controversial issues. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1105–1110.

Emmanuel Ebuka Ibeke, Chenghua Lin, Christopher David Coe, Adam Zachary Wyner, Dong Liu, Mohamad Hardyman Bin Barawi, and Noor Fazilla Abd Yusof. 2016. A curated corpus for sentiment-topic analysis. *Emotion and Sentiment Analysis*.

Hayate Iso, Xiaolan Wang, and Yoshihiko Suhara. 2021. Comparative opinion summarization via collaborative decoding. *arXiv preprint arXiv:2110.07520*.

Nitin Jindal and Bing Liu. 2008. Opinion spam and analysis. In *Proceedings of the 2008 international conference on web search and data mining*, pages 219–230.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Wei-Hao Lin, Theresa Wilson, Janyce Wiebe, and Alexander G Hauptmann. 2006. Which side are you on? identifying perspectives at the document and sentence levels. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 109–116.

Yang Liu. 2019. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*.

Mohammed Elsaid Moussa, Ensaf Hussein Mohamed, and Mohamed Hassan Haggag. 2018. A survey on opinion summarization techniques for social media. *Future Computing and Informatics Journal*, 3(1):82–109.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-first AAAI conference on artificial intelligence*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Stratos Xenouleas, Prodromos Malakasiotis, Marianna Apidianaki, and Ion Androutsopoulos. 2019. Sumqe: a bert-based summary quality estimation model. *arXiv preprint arXiv:1909.00578*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.