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 quickly understand relevant texts it gathered, 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 in a short time so that they can make the right purchase decision. This tool takes the raw reviews as input, separates the sentences and identifies those that praise the product (positive sentences) and those that criticize it (negative sentences). Then, it generates contrastive summarizations for the product: One for positive sentences and one for negative sentences each. Finally, it uses SUM-QE to evaluate the quality of the generated summarizations, which gives user the insight about how good the summarization is. Full code of this work could be found here1.

1 Introduction

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

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 (1) became a sensation as soon as it was proposed. Based on it, BERT (2) even became the SOTA model in a variety of tasks. In this work, we implemented Transformer-based models in three ways: First, we used BERT as a sentiment analysis model to classify Amazon raw product reviews into positive and negative clusters; Then, with the help of PEGASUS (3), a Transformer-based abstractive summarization model, we created contrastive summarizations for each product; Finally, we used BERT-based SUM-QE (4) to evaluate the quality of the generated summarizations.

We introduce some related work in §2, next, we elaborate the dataset preparation process (§3), generate summarization (§4). In §5, we evaluate the quality of the summarization and do some analysis on it. Finally, we conclude this work (§6).

2 Related work

In general, summarization techniques are divided to extractive summarization, which uses some of the sentences in the original text as summarization, and abstractive summarization, which paraphrases the key idea of the text in other words. There are many summarization model examples. (5) raised a RNN-based sequence model for extractive summarization of documents and show that it achieves performance better than or comparable to state-of-the-art. Then, after the appearance of BERT, we have seen its great application in extractive summarization (6). In the meantime, BERT is also widely used for abstractive summarization. (7) proposed a new fine-tuning schedule which adopts different optimizers for the encoder and the decoder as a means of alleviating the mismatch between the two, and (3) 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.

In this work, we focus on a specific kind of summarization: Contrastive summarization. The goal is to compare two opposing groups of texts and generate summarization for each group, in which the difference of the groups can be seen. A good pipeline can be found in (8), where they divide the contrastive summarization into three steps: Processing, semantic processing and summary creation. Different methods were summarized in the survey by (9). To add, expert-guided contrastive opinion summarization (10) took into account expert opinions and achieved better performance. Our work differs from these works in using Transformer-based models to enhance model performance.

There are many datasets available for contrastive summarization. The El Capitan corpus collected by (14) 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. Amazon product review dataset (13), on the other hand, 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 (15) collects over 1 million Tweets concerning Scottish Independence. However, this dataset only saves the Tweet and user ID, making it difficult to classify Tweets that support the Scottish Independence and that oppose it.

Apart from the short datasets mentioned above, some long datasets are also available for contrastive summarization. Bitterlemons corpus (16) 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 article. Another example is the COCOTRIP dataset (17), which contains 100 documents of trip reviews. Although these two datasets are well-designed for contrastive summarization, the number of texts is quite small. Therefore, we did not use them in the task.

In the summarization evaluation practice, ROUGE (11) is a widely used evaluation measure for summarization tasks. It compares the generated summarizations and expert-written golden annotated summaries. However, we have to switch to some other measures when the golden annotated summaries are not available. SUM-QE (12) provides 5 different aspects to measure the quality of summarization and does not need human-generated summaries. Also, it is proved this evaluation is highly correlated to human evaluation, making it quite suitable for this work.

3 Dataset Preparation

In this section, we introduce how we collect, clean and prepare the dataset for summarization. After we have done the preparation step, both positive sentences and negative sentences of one product will be saved separately and fed into summarization model in the next step.

3.1 Data source

Two similar datasets were used in this work: Amazon tagged product review and Amazon raw product review. The former dataset contains approximately 638 reviews, among which each sentence is annotated with sentiment; the latter contain over 320k reviews, no annotation is given. The both datasets are first collected by () and can be found here2. Note that one has to ask for permission for the access of the latter dataset.

3.2 Sentiment analysis

In this step, we used Amazon tagged product review to train a BERT-based sentiment classification model. Here, we split the reviews into sentences and delete 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 which are annotated with a “+” sign. We do not distinguish sentences which are only a little bit positive or extremely positive.
- **Negative:** Sentences which are annotated with a “-” sign. Same as positive ones, we do not care about the negative extend.
- **Neutral:** Sentences which 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 “BERT-large-uncased” model from Huggingface with different batch sizes (see Table 1). In the evaluation of 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 regarded as correct as long as it is not classified as the opposite sentiment, which means the classification of sentences whose ground truth is neutral is always regarded as correct. The reason behind this idea is simple: It does not matter if the positive sentence group (or negative sentence group) contains some neutral sentences as they are very likely not to influence the summarization output. Therefore, it is enough to achieve a high relaxed score in this task.

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 the batch size equal to 8. Although the model with batch size equal to 8 wins this model in epoch 4 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 the fact 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 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 is stated above, there are quite many reviews in the dataset, but we do not need all of them. Hence, in this work, we only chose those products with between 100 and 200 reviews available. Then, we split them into sentences and predict the sentiment. Finally, we only keep the sentences with the confidence score more than 0.95 and saved them to txt files. Each products have two files: Positive and negative.

4 Summarization

In this step, we intended to use an abstractive summarization model to extract the most important information in the reviews. The reason why extractive summarization is not suitable for this task is that 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 tested five PEGASUS variants with the sentences from Section 3 (“google/pegasus-xsum”, “google/pegasus-large”, “google/pegasus-cnndailymail”, “google/pegasus-multinews”, “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 dataset. As is shown in Table 2, their performances vary quite much. For “google/pegasus-xsum”, the summarization is quite short, and even worse, it generates a positive summarization for negative sentences; for “google/pegasus-cnndailymail”, all of the sentences are extracted from reviews, which makes it behave like an extractive summarization model; the situation is even worse for both “google/pegasus-multinews” and “google/bigbird-pegasus-large-arxiv”: They generate nearly unreadable texts and nearly do not convey any useful information. Compared to them, “google/pegasus-large” performed quite well with respect to readability as well as 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 (12) to evaluate the quality of generated summarizations. SUM-QE stands out for not requiring human-annotated summaries as golden summaries, 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.

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

- **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 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.

For each quality measure, if the score of the generated summarizations falls in [average - std, average + std], we call it a fair score; If the score is above this range, we call it good; if the score is below the range, we call it bad. The example in Table 4 shows the quality difference between a good, fair and bad summarization.

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

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 examples (Q4)

To recap, Q4 measures focus. The good example is quite fluent to read, stating this product is much better than the former one and listing the pros clearly. The fair example, however, praises the product for its good sound, but starts to talk about battery in the end. Finally, in the bad example, multiple topics can be seen, such as usability, good monitor and reliability.

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 then use this on the larger Amazon raw product review dataset to classify review sentences to positive and negative ones. We then put these sentences into PEGASUS model to generate abstractive summarization for each product. With SUM-QE model, we obtained the quality scores of the summarizations in different aspects and analyzed the result. According to the analysis, we separated the summarization to three different classes: good, fair and bad. We believe this is an important proxy when consumers read the summarizations and make shopping decisions.

7 Future work

Possible future work can focus on improving this tool by using Amazon API to automatically obtain product reviews. So far, one needs to enter the product reviews manually in order to generating the summarization. For some popular products, there might be hundreds of reviews, 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 the future work can be improving summarization model. So far, the model only takes English as input, which makes the tool limited for non-English speaking users. Training a multi-language abstractive summarization can solve this problem.

Finally, the size of the whole model is quite big 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.

Acknowledgements

I would like 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.