Nuance in Headline Generation: Comparison of Performance of TextSum and TextRank on a Dataset with Sentiment

Riya Mokashi
ram882@nyu.edu
Juraj Jursa
jj2664@nyu.edu

Abstract

This paper attempts to compare the performance of extractive and abstractive summarization on a dataset that consists of different media outlets describing the same event. The goal of this paper is to determine, how well can TextSum, an abstractive summarization model, pick-up sentiment in a text summarization task of generating a headline for a given document, compared to an extractive TextRank. Both models were evaluated on their ability to not only accurately match the headlines given, but also to match the sentiment displayed in the articles.

1 Introduction

Text summarization attempts to generate a short version of the document provided, while retaining the most important information. Headline generation is unique in that it introduces the need for not only an extremely concise summarization of text, but it also requires that the content cover of the summary is broad and ”compelling.” Due to the influx of information available online, automatic machine-generated headlines are proving to be useful. There is much to benefit from being able to reduce massive amounts of information down to their relevant takeaways.

There are two main ways to approach text summarization which were both considered for this analysis. The first is extractive summarization which extracts the most relevant sentences from a passage and poses those as the summary. However, it is extremely rare that any one sentence in a news article will be able to perform as a concise yet in depth summary of the article at hand. The second type of is abstractive summarization, sometimes called generative summarization. Generative summarization, on the other hand, generates an original summary with original sentences (not already seen in the document).

This then brings us to sentiment matching. Many news articles have bias in both information and sentiment. These articles will also often have headlines that reflect this subtlety in sentiment change. Ideally, a headline generation model will be able to not only capture the information being delivered, but also the sentiment that is trying to be conveyed.

While both TextRank and TextSum models perform reasonably well on a neutral dataset such as Gigaword, we explore whether one of them should be preferred for a task of text summarization, where the context is not necessarily neutral. For these purposes, we have decided to take Liu and Pan [1] model, which has made their pretrained seq2seq model available to use, as well as TextRank by Mihalcea and Tarau [2], which is the state of art for extractive summarization. Liu and Pan TextSum included their baseline results which are evaluated on a ROUGE metric, which we will use in our paper as well. Our hypothesis is, that given the simplicity of TextRank and the lack of generative component, it is most likely going to focus on the words that convey the most content, rather than the sentiment. We expect TextSum to fare better on this task.

2 Text Summarization and Sentiment

Text summarization is a task, that mostly focuses on an accurate description of a paragraph. The issue that we are trying to explore is not necessarily just the accuracy of the description, but rather the question of how well do the text summarization models also convey the sentiment that the paragraph has. Large amount of research has been dedicated to the exploration of how to tune performance, but very often, these models were trained on a dataset that contains largely neutrally worded content. With the dataset that we have available, we are trying to look at whether the output also matches the tone, the positive or negative feelings injected into the description of the event at hand.

Sentiment analysis is one of trivial tasks that vast majority of researchers in this field have encountered during their studies. What we are trying to explore here, is whether text summarization models have to an extent learned the subtleties of sentiment along the primary task of text summarization. Summarization as such should not only convey the factual information from the input text, but also the more nuanced content of the original writer's opinion on the topic at hand. While models like BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [3] attempt to solve this necessity for nuance by using bidirectionality, we would like to see, whether off-the-shelf extractive and abstractive summarization models are able to learn the general sentiment conveyed and change the headline accordingly.

3 Baseline models

TextSum is an abstractive summarization model. At it's core it is a traditional sequence-to-sequence model with attention. TextSum does need to be trained and will be trained on the English Gigaword dataset.

TexRank is a graph-based NLP model, which does extractive summarization based on the analysis of importance of individual vertices which represent words in an article. The number of edges connected to the respective vertex determines its importance. The relevance of connected edges is recursively determined by the indegree of the vertex from which the edge is coming. This is inspired by PageRank that Google uses to determine the recommended webpages in their search engine.

We specifically chose these two models because we wanted to look at both abstractive and extractive summarization models that could provide benchmark performance levels for the capture of sentiment in summaries/headlines. Through this we also aimed to test a coarse to fine testing of headline generation similar to the study done by Tan et. al (2017) [4]. Extractive summarization is very coarse in that it often has difficulties picking up nuances due to the fact that it is limited only to the sentence available to it in the text. On the other hand seq2seq models, specifically TextSum have had proven performance on headline generation on the past.

Just off of the two baseline models, we hypothesized that TextSum would have better performance when it came to matching the headlines at hand but TextRank would have superior performance in sentiment capture. This is because TextRank would pull phrases from the article at hand which is more likely to contain sentiment conveying words.

4 Dataset

4.1 Training data

TextSum is trained on the English Gigaword dataset from the Stanford Linguistics department. This dataset as a whole contains several years of articles from 6 major news agencies and each article comes with a clearly delineated headline and text. Once we completed preprocessing the data we were left with over 5 million news articles with a grand total of over 200 million words. TextRank on the other hand is a completely unsupervised model which requires no training data.

4.2 Sentiment dataset

Our dataset contains over 7000 articles on a topic of ”Belt and Road Initiative”, which is a global development strategy of Chinese Government, which involves investment into infrastructure in almost 70 countries for trading purposes, renewing the historical ”Silk Road” that was an integral part of the trading infrastructure for the better part of the last two millennia. The dataset has been scraped from the Factiva database and encompasses articles from 2015-2020.

The reason for using the BRI data as our testing dataset is because the dataset introduces two new obstacles to the models.

The first is that the datasets encompass a very small scope of news as they all focus of the Belt Road Initiative. This is similar to work done by Xue et al. [6] who looked at the effects of topic sensitive headline generation. We will be taking a similar approach where we want to test the nuance in the headlines generated. Since we have several articles from different regions and sources each discussing the same event, we have slightly different aspects of the same event being reported. In turn, the accuracy of the generated headlines will be incredibly important.

The second obstacle is that these articles will be coming from three major groups - Cooperating countries, non-cooperating countries, and China. This gives us a spectrum of sentiment that will be displayed throughout the dataset. This introduces nuances to what data needs to be displayed in the headline. For a headline generation model to perform well on this dataset, it will need to be able to accurately capture the sentiment depicted in the articles.

Due to the majority of them being quite neutral on this topic, we have decided to pick around 2500 of them, where 1000 conveys a positive sentiment, 500 negative and 1000 neutral, to use data which has more sentimental outliers.

5 Experiments

5.1 Evaluation Metric

There are two key things we will be evaluating throughout this experiment. One is whether the generated headlines accurately match up to the original headlines. This will allow us to measure how well the models are working on understanding and pulling data from the articles at hand. The other thing we are evaluating is how well the sentiment is captured between the article and the generated headline.

The performance of the original TextSum model as well as TextRank was measured using the ROUGE evaluation metric. As discussed in the paper by Lin [5], this metric measures the recall and the accuracy of the generated output compared to the gold standard (the label), along with n-gram overlaps and the longest matching word sequence. Due to this, we decided to compare their performance on our dataset on ROUGE as well.

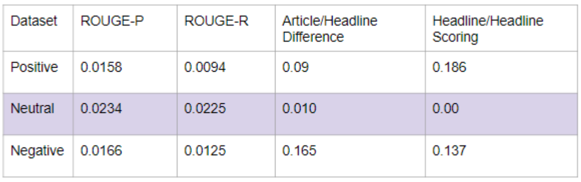
ROUGE will be the metric for evaluating the generated headlines vs the actual headlines. Since this is the standard metric in multiple other papers as well it allows us to also extrapolate our results to consider how the models perform in their usual environments (standard Gigaword dataset)

For the sentiment analysis we will be looking at a simple absolute difference in sentiment scores. We will be measuring the difference in sentiment between the article and the actual headline vs the article and the generated headlines. This is because we want to account for cases where the headline may have significantly more polar sentiment than the article at hand due to sensational headline tactics.

5.2 Process

The data was first processed and cleaned at which point we used the TextBlob dictionary to assign sentiment scores to each of the articles. The main thing we found was that the overwhelming number of articles were neutral, in fact an even larger percentage had an exact 0.0 sentiment score. Once the scores were set we then separated the articles into 'Positive', 'Neutral', and 'Negative' sentiment articles. We then sampled 1000 of the most positive articles, 1000 of the most neutral, and 500 of the most negative. We had to restrict the number of negative articles due to the fact that it was difficult for the sentiment dictionary to accurately pick up negative sentiment in the articles and there were very few articles that were classified as negative. The TextRank model was parsed for each of the articles and was restricted to outputting a 15 word max summary which would act as a defacto headline. The TextBlob sentiment dictionary in Python was then used to provide a rough sentiment score for the articles, the actual headlines, and the generated headlines. For the TextSum model we trained it on Gigaword with the intention of then testing the model exclusively on the BRI dataset we had on hand. However, due to time and equipment constraints we were unable to get the Textsum dataset trained and working in time. Due to this we will only be going into the TextRank model in the results section.

5.3 Results



Dataset	ROUGE-P	ROUGE-R	Article/Headline Difference	Headline/Headline Scoring
Positive	0.0158	0.0094	0.09	0.186
Neutral	0.0234	0.0225	0.010	0.00
Negative	0.0166	0.0125	0.165	0.137

When evaluating the TextRank outputs we found that the model was actually performing quite poorly when just producing the headline. The neutral sentiment labeled articles saw an increase in performance which can largely be attributed to the fact that these articles were largely informative and so were the corresponding headlines. The neutral labeled articles were also the best at matching the sentiment of the original headlines. There was a slight difference in sentiment at about 0.003 which was lost in the rounding process. We have reason to believe that this set of results could be misleading. There are several explanations for why this could be which will be discussed in the next section.

5.4 Discussion

The ROUGE score was quite low for TextRank model. The likely reason is the already mentioned shortcomings of an extractive summarization that were mentioned in the introduction. The other, more fundamental problem, with our experiment is that even though the article may be quite biased towards either side, the headline does not necessarily conveys that, and the nuance is only shown in the article itself. Despite the fact that we did not manage to get TextSum work in time, this problem is relevant for both extractive and abstractive summarization, as it renders the ROUGE score imprecise because it compares the generated headline to the original one, which might not be biased in the first place. This means, that instead of looking for a high ROUGE score necessarily, the examination should focus on the difference in ROUGE score between our dataset and Gigaword corpus or other similarly neutral dataset. This would show, that the model is reacting and behaving differently if the examined article is biased, and then would require subsequent examination by potentially using a sentiment analysis model. This type of closer scrutiny could be a worthwhile path to take as a follow-up for this type of project, one which would however require a more advanced evaluation metric than ROUGE.

6 Collaboration Statement

Riya worked on executing the models, sentiment analysis, and retrieving the evaluation results. She also wrote the Experiment and Baseline Models sections as well as contributing to other sections. Juraj wrote the introduction, chapter on text summarization and sentiment, chapter on datasets, and the evaluation metric and discussions in the experiments chapter. Both of us worked on editing and reviewing the final work. The code can be found at github.com/RiyaMokashi/MLLUHeadlineGeneration

7 References

[1] Liu P. Pan X. (2016) Text Summarization with Tensor Flow. https://ai.googleblog.com/2016/08/text-summarization-with-tensorflow.html

[2] Mihalcea R. Tarau P (2004) TextRank: Bringing Order into Texts. In ACL 2004, pages 404–411, 2004.

[3] Devlin, J., Chang, M., Lee, K., Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv, abs/1810.04805.

[4] Tan, J., Wan, X., Xiao, J. (2017). Recent Advances on Neural Headline Generation. JournaFrom Neural Sentence Summarization to Headline Generation:A Coarse-to-Fine Approach

[5] Lin, C. ROUGE: A Package for Automatic Evaluation of Summaries

[6] Xu, L., Wang, Z., Ayana, Liu, Z., Sun, M. (2016) Topic Sensitive Neural Headline Generation. ARXIV abs/1608.05777v1

[7] Liu Y. (2019) Fine-tune BERT for Extractive Summarization. ACL.

[8] Michele Banko, Vibhu O. Mittal, and Michael J. Witbrock. Headline generation based on statistical translation. In ACL, pages 318–325, 2000.

[9] Carlos A. Colmenares, Marina Litvak, Amin Mantrach, and Fabrizio Silvestri. HEADS: headline generation as sequence prediction using an abstract feature-rich space. In NAACL, 2015.

[10] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In EMNLP 2015, pages 379–389, 2015.

[11] Rui Sun, Yue Zhang, Meishan Zhang, and Dong-Hong Ji. Event-driven headline generation. In ACL 2015, pages 462–472, 2015.

[12] Ayana, Shen, S., Lin, Y., Tu, C., Zhao, Y., Liu, Z., Sun, M. (2017). Recent Advances on Neural Headline Generation. Journal of Computer Science and Technology, 32, 768-784.

[13] Al-Sabahi, K., Zhang, Z., Yang, K. (2018). Bidirectional Attentional Encoder-Decoder Model and Bidirectional Beam Search for Abstractive Summarization. ArXiv, abs/1809.06662.

[14] Lin, E., Chiou, D., Vasanth, S. Tang, W. (2018) News Headline Generation