An Exploration on Summarization and Outlining of Persuasive Arguments Using Recursive Neural Network

Yu Tian[*]
Northeastern University
tian.yu2@husky.neu.edu

Justin Koser[†]
Northeastern University
koser.j@husky.neu.edu

Yifeng Mao[‡]
Tsinghua University
mao.yife@husky.neu.edu

Abstract

Our research is mainly focused on the topic of auto summarization of long texts. We have already finished the building of the basic model in our mid-report, and we have applied atention model and bi-directional LSTM into our final project. Basically, we have improved the performance of existed models for Although the result is not produced as expected, we still analyze the reson for our failure and try to give out the possible improvement for our model.

Keywords Automatic Summarization, Seuqence to sequence, Recurrent Neural Network

1 Introduction

In this project, we are trying to build up a program for auto-summarization based on the seq2seq model, which is a built based on idea of recurrent neural network. Word embedding, sequence to sequence model and result evaluation are the three steps to generate a summary in our method. For word embedding part, we utilize the pretrained word embedding model, GloVe. For building the recurrent neural network, we take an encoder-decoder model for the basic structure of sequence to sequence model, and we apply some improvements on the basic model to enhance the performance such as implementing bidirectional lstm cell and adding attention mechanism. BLEU and ROUGE, the state of art evaluation metrics, are employed for results evaluaton.

2 Related work

Auto summarization is one of the most challenging tasks in natural language processing, and there are many people have investigated previously. Basically, we divide auto-summarization into two general approaches, extraction and abstraction.

Extraction methods extract sentences directly from the original text to generate the summary. Luhn et al. [3]introduced a method to select representative sentences of a document depending on term frequency, while ignoring meaningless words, like preposition, that are common across the language. Edmundson et al. [4]introduced a model based on key phrases. In addition to standard frequency depending weights, this model also used cue methods, title methods and location methods.

Meanwhile, abstraction methods generate a summary by using semantic representation. This method can generate words that are not included in the original documents. This method is able to understand and explain the meaning of the texts by using its own words. Recently, Ganesan et al. tried generating summaries with the application of the abstractive method based on phrase analysis.

WangâĂŹs[5] work, one of our important reference, introduces neural networks to auto summarization, which do not need human-annotation for summarization tasks.

3 Dataset

We used two datasets to train and evaluate our model. The first dataset is originally from Idebate website.. Each debate proposition has lists of supporting and opposing claims, which we regard as the summarization of the texts. Wang and Ling has collected the original version of the corpus, which we use to train our model at first. In order to improve the robustness of our model by feeding more data into it, we have collected more data from the website with our data crawler.

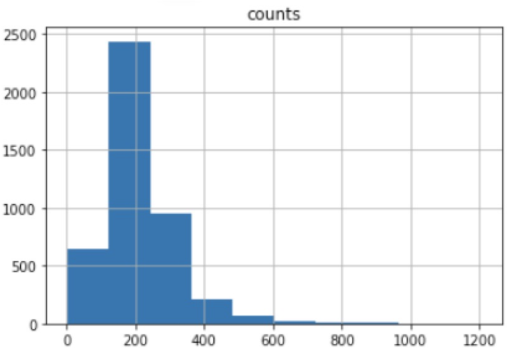


Figure 1. Distribution of text length of Idebate texts

[*] Second year graduate student from data science

[†] Second year graduate student from computer science

[‡] Junior student from automatization department

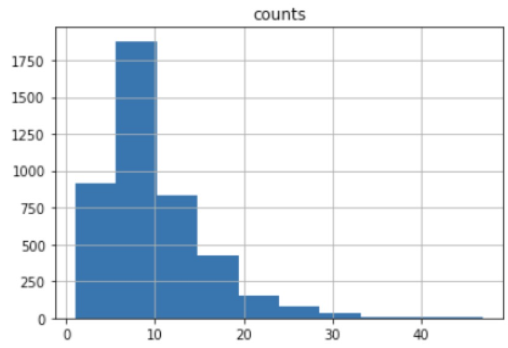


Figure 2. Distribution of summary length of Idebate summaries

Another dataset is Amazon food review on Kaggles. This dataset consists of reviews of fine foods on Amazon website from 1999. The data includes all 500,000 reviews up to October 2012. The reviews contain product and user information, ratings, and a plain text review. The food reviews also include reviews from all other Amazon categories.

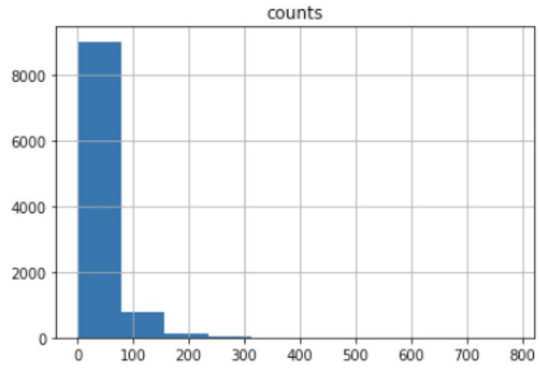


Figure 3. Distribution of text length of Amazon food review

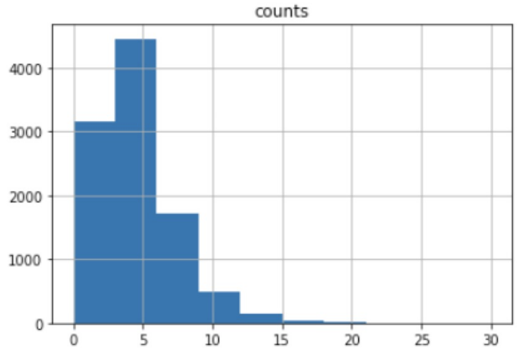


Figure 4. Distribution of summary length of Amazon food review

4 Methodology

4.1 Basic seq2seq model

Our basic sequence to sequence model uses text sequence as input, and try to predict summary sequence as output. Encoder and decoder are two independent set of RNN, while each rnn is two layers of LSTM cells. The hidden nodes in one layer is 256. Encoder RNN and decoder RNN are connected by passing encoder state into decoder. Because we use two datasets for training and predicting, we train two models respectively. But the size of two datasets are around the same, each of then includes approximately five thousand pairs of text-summary.

4.2 Bidirectional seq2seq model

Bidirectional seq2seq model applies bidirectional RNN to encoder, in order to extract the contextual information from the text. The principle of Bi-RNN is to split the neurons of a regular RNN into two directions, one for positive time direction, and another for negative time direction. By using two time directions, input information from the past and future of a certain time frame can be used. As for decoder, we still use standard RNN.

4.3 Attention seq2seq model

Attention seq2seq model introduces attention Mechanism to decoder cells. Attention means assigning greater weights on certain words, and these words are more likely to be picked out as summary. By using attention wrapper API in the tensorflow, we are able to use encoder output which contains rich contextual information.

5 Experiments

5.1 Experiment setup and data preprocessing

For the implementation of our Sequence to Sequence model, we chose to use tensorflow 1.7 version API as our main deep learning framework. All scripts are written in python 3.6. We implemented two kinds of models: bi-directional lstm RNN, and bi-directional lstm RNN with attention mechnism.

For data preprocessing and experiment setup part, we introduced word embedding and padding. Word Embedding represents a series of methods transferring text words to quantity vectors[1]. For our project, we are using pre-trained GloVe word vector offered by Stanford University to embed our input data[2]. Basically, GloVe is an unsupervised machine learning to generate vector representations for words. For our program, we build a word list, and replace every word in the text and summary with their index in this word list, which will be used as search index in future embedding. Meanwhile, we build a list of word vectors with the corespondding order to the word list, which performed as our look up table. It is worth mentioning that, for decoder input, we have some special processing. First, we add "<GO>" sign as the signal for start of decoder input, "<EOS>" as the end of the sequence and let "<PAD>" make up other positions if the processed sequence is still shorter than our preset decoder length. The reason why we do this is shown in figure 1 attached as appendix. For training step, we feed text as input into encoder as well as the summary as input for decoder. However, for prediction step, we only have text to feed into encoder without anotated summary. So for decoder, we take encoder state as initial input, and set the prediction result from each decoder cell as the input for next decoder cell. When we detect <EOS> sign in our prediction result, we output our prediction sequence as the final summary for a given text.

5.2 Model training on IDBATE dataset

At first, we applied these three model on IDEBATE dataset.After tuning the model for setting hyperparameter, we set the hyperparaters as following table.

learning rate	0.001
RNN size	256
Optimizer	AdamOptimizer
Number of layer	2
Drop out rate	0.2

During the test, we found the model does not work well on our own idebate dataset. LSTM cannot capture as much information as we want given long text. Even we add bidirectional and attention mechanism, the result seems does not make a lot of sense. Following graph shows the validation loss over batches during the trianing process. The loss drops quickly at first but stop dropping at around 2.

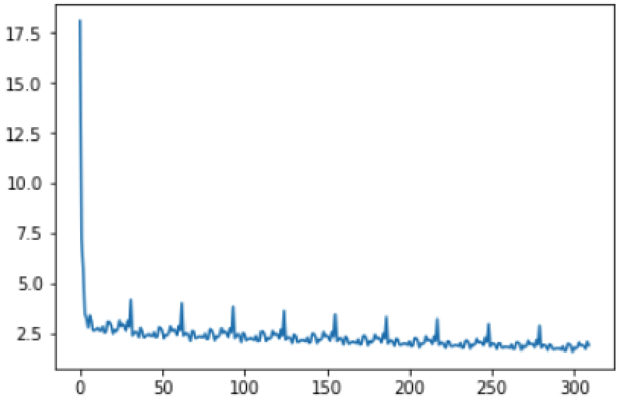
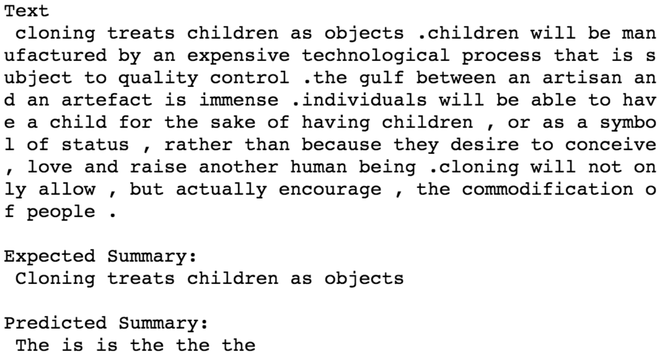


Figure 5. Validation loss during training attention model on IDEBATE

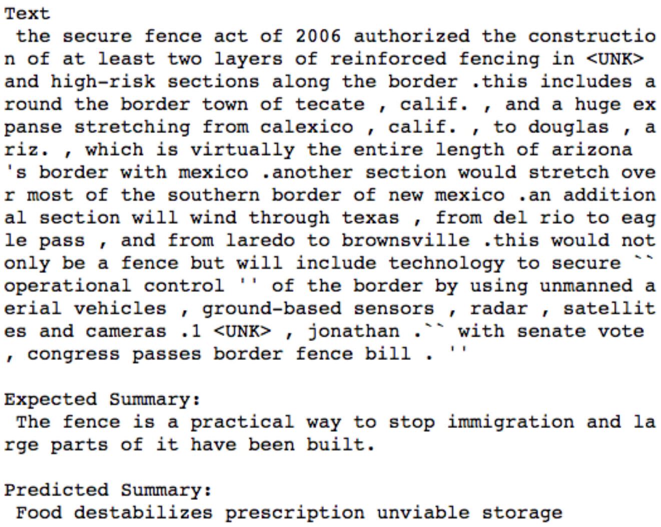
Following graph shows the sample predicted result for produced by bidirectional lstm rnn model. According to the results, the predicted summaries are basically the most prequenty words in the corpus.



```
Text
 cloning treats children as objects .children will be man
ufactured by an expensive technological process that is s
ubject to quality control .the gulf between an artisan an
d an artefact is immense .individuals will be able to hav
e a child for the sake of having children , or as a symbo
l of status , rather than because they desire to conceive
, love and raise another human being .cloning will not on
ly allow , but actually encourage , the commodification o
f people .

Expected Summary:
 Cloning treats children as objects

Predicted Summary:
 The is is the the the
```

Figure 6. Predicted results based on IDBATE and bidirectional model

However, after the attention mechanism are added, the summaries return from corpus seems to be a combination of a small subset of overall vocabulary. Sample output is shown as following.

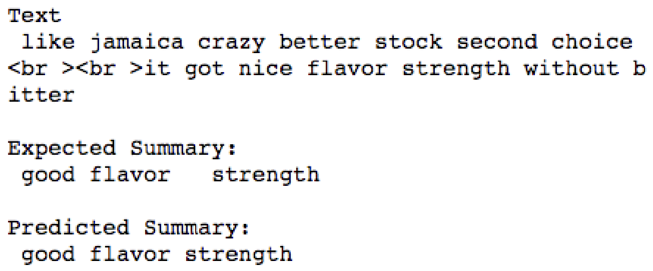


```
Text
 the secure fence act of 2006 authorized the constructio
n of at least two layers of reinforced fencing in <UNK>
and high-risk sections along the border .this includes a
round the border town of tecate , calif. , and a huge ex
panse stretching from calexico , calif. , to douglas , a
riz. , which is virtually the entire length of arizona
's border with mexico .another section would stretch ove
r most of the southern border of new mexico .an addition
al section will wind through texas , from del rio to eag
le pass , and from laredo to brownsville .this would not
only be a fence but will include technology to secure ``
operational control '' of the border by using unmanned a
erial vehicles , ground-based sensors , radar , satellit
es and cameras .1 <UNK> , jonathan .`` with senate vote
, congress passes border fence bill . ''

Expected Summary:
 The fence is a practical way to stop immigration and la
rge parts of it have been built.

Predicted Summary:
 Food destabilizes prescription unviable storage
```

Figure 7. Predicted results based on IDBATE and attention model

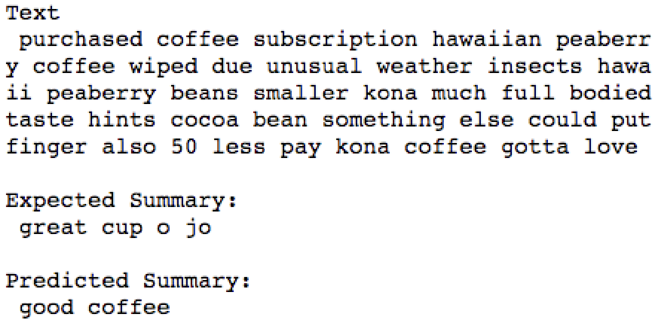
5.3 Model training on Amazon Food Review dataset

For the sake of comparision, we tried to train our models on shorter amazon food review dataset using models of bidirectional-lstm with and without attention mechanism. The results seems more resonable this time. Sample output are showed as following.



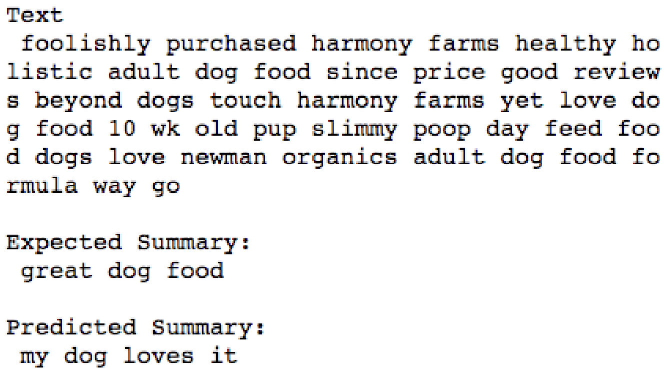
```
Text
 like jamaica crazy better stock second choice
<br ><br >it got nice flavor strength without b
itter

Expected Summary:
 good flavor   strength

Predicted Summary:
 good flavor strength
```

Figure 8. sample1 for amazon food review



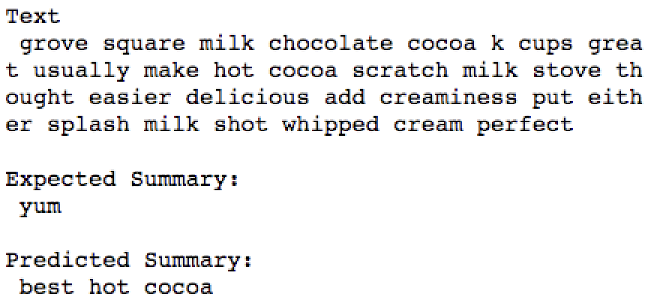
```
Text
 purchased coffee subscription hawaiian peaberr
y coffee wiped due unusual weather insects hawa
ii peaberry beans smaller kona much full bodied
taste hints cocoa bean something else could put
finger also 50 less pay kona coffee gotta love

Expected Summary:
 great cup o jo

Predicted Summary:
 good coffee
```

Figure 9. sample2 for amazon food review



```
Text
 foolishly purchased harmony farms healthy ho
listic adult dog food since price good review
s beyond dogs touch harmony farms yet love do
g food 10 wk old pup slimmy poop day feed foo
d dogs love newman organics adult dog food fo
rmula way go

Expected Summary:
 great dog food

Predicted Summary:
 my dog loves it
```

Figure 10. sample3 for amazon food review



```
Text
 grove square milk chocolate cocoa k cups grea
t usually make hot cocoa scratch milk stove th
ought easier delicious add creaminess put eith
er splash milk shot whipped cream perfect

Expected Summary:
 yum

Predicted Summary:
 best hot cocoa
```

Figure 11. sample4 for amazon food review

Table 1. Performance Scores

Method	ROUGE_1	ROUGE_2	BLEU
Ref:text Sum:human	0.099	0.024	-
Ref:text Sum:auto	0.059	0.008	-
Ref:human Sum:auto	0.189	0.071	6.176

5.4 Evaluation

Since our models trained on Idebate doesnt́ make sense at all. So, we would like to implement evaluation on the results of amazon food reviews first, and then we will try to show the results of Idebate.

We first consider the ROUGE score of human summary and original text as the best result for auto summarization. We take two measurements in ROUGE score: ROUGE-1, ROUGE-2. And we apply three method using each ROUGE metric. The first method considers original text as the reference of the summarization, while regards the human summary as the summarization result. We use this method to find out the baseline ROUGE score of the summarization task. Then, the result of auto summarization is evaluated with the reference of original text first. Last, we would like to show the ROUGE score of two summarization in the third method. We list ROUGE and BLEU score in the following chart.

From the table above, all the ROUGE scores are small numbers. The reason for the small scores is the length of the summary is realtively short, and ROUGE metrics have better performance in comparing the similarities of long sentences. We can see that the gap between ROUGE-1 and ROUGE-2 scores in all the three methods. Since the summary is short, either computer generated summaries or human summaries are difficult to catch the same bi-gram in the text.

Thus, we use BLEU to better explain our summarization result. Although BLEU metric is designed for evaluating auto translation, we can regard the auto summaries and human summaries has the same meanings. To some extent, they are able to translate each other. The BLEU score shows that our summarization is able to produce satisfying result.

5.5 Error analysis

It is obvious that basic encoder-decoder lstm rnn structure does not work well on long text summarization. Even though the attention mechanism and bidirectional cell are implemented, which are proved to lead significant improvement on rnn model, results are improved limited for the case of long text summarization. For the short review text, the model obviously learn some weak concept such as good, bad, tasty. However, for the short text summarization, the problem seems more likely to be an information extraction instead of summarization problem. Result often contains the key words along with the sentiments. Besides, since the original summarization for food reviews are very similar. It is more possible that the rnn memorized some of the common summaries. When it comes to the reviews for IDBATE database. The average length of the text is above 200 words. The lstm rnn model becomes very weak on learning pattern by directly feed text inputs into the model. Protential improvement may be conducted by adding structural information such as the position of the sentences, or combining extractive method with abstractive methods.

6 Conclusion

In our project, we try to convert text summarization into sequence to sequence task, and we have some positive results on the Amazon food review. Although our model is difficult to generate accurate summaries on the Idebate datasets, we still try to figure out the reasons for that. In the future, we are going to focus on the structure of the text, and improve the perfomance of our model on the Idebate summarization.

References

[1] https://en.wikipedia.org/wiki/Word_embedding

[2] https://nlp.stanford.edu/projects/glove/

[3] Peter Luhn. 1958. The automatic creation of literature abstracts. IBM Journal of research and development 2, 2 (1958), 159-165.

[4] Harold P Edmundson. 1969. New methods in automatic extracting. Journal of the ACM (JACM) 16, 2 (1969), 264-285.

[5] Lu Wang and Wang Ling. 2016. Neural network-based abstract generation for opinions and arguments. arXiv preprint arXiv:1606.02785.2