Movie Comments Generation and Grading with topic-based GRU

Ding Pan
517021910871
Automation, SJTU
huanmie@sjtu.edu.cn

Jiasheng Xu
517021910958
Automation, SJTU
xujiasheng@sjtu.edu.cn

Bo Yue
517021910825
Automation, SJTU
yuebo2017JD@sjtu.edu.cn

Abstract

In this paper, we focus on movie comments generation, which is a challenging Natural Language Generation(NLG) task that generates a comment with multiple topics. There are some existing difficulties towards understanding different topics and enriching comments' expressions. To solve the problem, we develop two topic-based models and make comparisons between them. One is to simply input the topics. The other is to learn how to encode the topic through attention mechanism to get a weighted-average topic and use it as the input. Our BLEU[1] score of the result in test set is relatively high compared to empirical baselines. Further, based on the model we have just trained, we slightly alter the model to accomplish another task, whose purpose is to evaluate the emotion of the comment. Based upon the parameters of the previous model instead of initializing from scratch, the learning speed is far faster than that without previously-learned parameters. We make comparisons between models under different situations. For the second task, we also achieve a high accuracy in classifying comments' emotions.

1 Introduction

Natural Language Generation(NLG) is a fundamental research and challenge in the field of Natural Language Process(NLP). NLG aims to generate high-quality natural language text just like that of human-beings. It's commonly known that, NLG is a core technique widely applied in machine translation, text summarization and question-answering system, etc. Many effective models concerning the combination of NLP and neural networks have already been introduced, such as Recurrent Neural Network(RNN) and Long Short-Term Memory Networks(LSTM). In this paper, we choose to use a more efficient network, called GRU, which takes topics as inputs and outputs a movie review under the theme semantics of the topics, to finish the movie review generation task. Besides, to further extend our model for application, we also create a second network which can score the human reviews as well as our generated movie reviews. In Fig.1, we show a simple example of review generation and score procedure. Our two models can be applied in various scenarios to reduce human workload. For instance, we can generate a general movie review about a certain movie instead of spending time organize our words to give out a review. Whst's more, the model can provide a recommended score to the movie when a human-written review is put onto the reviewing website. This really manifests the progress of Artificial Intelligence, as writing skills have long been mastered by human beings alone.

In terms of our first model, we model the review generation process in neural networks. Afterwards, we make comparisons between this model and an extended model into which the attention mechanism is fused. The former model stresses n-gram coherence while the latter one indicates paragraph-wise coherence or main idea. These two aspects concerning coherence are also main challenges in NLG nowadays, which need to be solved properly. As for our second model, since we already relate words with vectors in our first model, we just add two more layers to the existing models to grade the input reviews.

[1]BLEU (Bilingual Evaluation Understudy) is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another.

In this paper, specifically, we represent the topics by mapping the topic words into embedding space with a dimension of three hundred and utilize GRU as generator. During the generation process, the attention mechanism is applied at each time step in order to build semantic relationships between topics and generated words. When it comes movie reviews, we choose two reasonable topics, the movie name and the given star(for each movie, there are five stars representing the degree of the watcher's positive feelings towards the movie).

Unfortunately, as far as we three have researched, there is no existing public yet large-scale dataset for our review generation. Therefore, we opt to produce dataset on our own. In order to test the effectiveness of our approach, we build a review dataset which is directly based on a famous movie reviewing website by means of web crawler techniques. At last, we compare our models with relevant models in terms of metrics, like BLEU, etc. Also, a comprehensive evaluation and future work to be done is also included in the paper.

2 Data Acquisition and Processing

We describe the proposed approach for data acquisition and processing in this section. Douban[2] is a famous website in China which provides information about movies, literature and musics. On the website, in particular, there are a large number of reviews as well as scores(stars) of the movies graded by the reviewer which provides people with movies that are worth watching.

Therefore, we choose Douban as the source to acquire the needed data set. We employ the web crawler technology to extract relevant contents about movies by utilizing the naturally annotated web resources on Douban. After that, we process the data in a series of steps in order to make the data suitable to be trained in the neural network.

2.1 Data Acquisition

In order to acquire the movie name, stars(people score each movie from 1 star to 5 star) and reviews of each movie, we build up a python crawler. For a certain movie, there are three levels of reviews, which are good, common and bad. By logging in Douban, we can acquire at most 500 reviews for one kind of review of a movie. What's more, in case unreadable foreigners' names are hard to understand and effect the result, we mainly focus on Chinese movies. We randomly choose reviews from 20 movies as our data set. The raw data acquired is in this form: (review, star, movie name). Two examples are shown in Table 1.

Table 1: Raw Data Form

review	star	movie name
It makes me laugh.	4	Shaolin Soccer
What a trash!	1	Shaolin Soccer

2.2 Data Processing

After acquiring the raw data, what we have to do is to process the data so as to make it fit the requirements of input format of the neural network. Our process of dealing with the raw data is listed as follows.

First of all, some low-quality reviews should be removed. To be more specific, we assume reviews whose length are less than 6 Chinese characters not well written, thus deleted. Additionally, reviews with special symbols(e.g. a smile face emoji) should also be removed in case they cannot be found in the word vectors. We hope the reviews of every star of a certain movie(range from 1 to 5) are equal in terms of quantity. Thus we randomly choose 100 reviews for each (movie name, star) and obtain approximately 10000 reviews in total.

Secondly, unlike English sentence, the acquired reviews is composed of separate Chinese words, we need to divide every sentence into a set of words, where we simply apply jieba[3] package to cut the sentence into words. Thus, we reasonably find a pre-trained 300-dimension word vectors(Word2vec) encoder which encodes every word into a 300-dimension vector. If the word is not included in the word vectors, we first split it into single characters and encode them in the same way. However, if the single character is still not in the word vectors, we just ignore it.

At last, we conduct one-hot encoding to the movie name instead of using the word vector encoder for the reason that movie names may have different meanings with each Chinese characters in them. For the star of the movie, we simply

[2]Douban.com, launched on March 6, 2005, is a Chinese social networking service website that allows registered users to record information and create content related to films, books, music, recent events, and activities in Chinese cities.

[3]jieba is a Word2vec library, widely applied to divide sentences into words

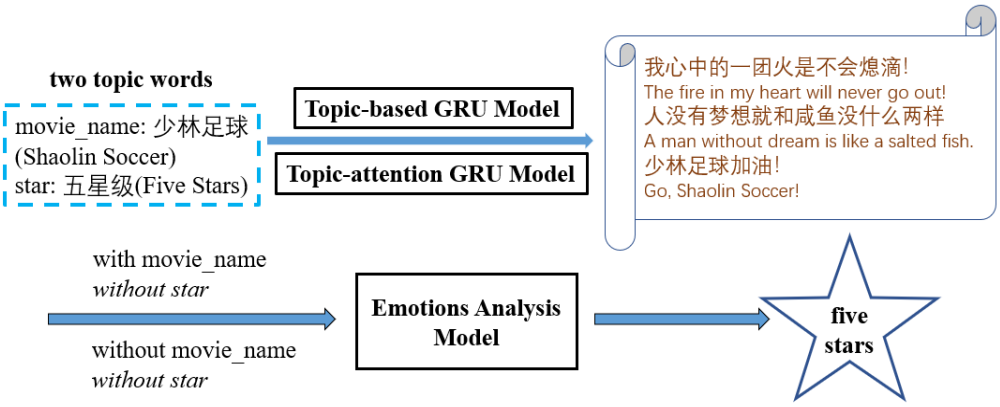

Figure 1: The basic flowchart for our paper.

transfer it to the Chinese character and encode it by the word vector.

Now we attain the vector form data. The whole process can be seen in Fig.2

3 Comments Generation Models

In this section, we describe the proposed comments generation models. Comment Generation Models aims at generating movie comments through two topic words, which are movie name and corresponding star.

In terms of our Natural Language Generation(NLG) models, we have adopted two major models based on topics. Topics here simply mean combinations of each movie name and its corresponding star($star \in \{1, 2, 3, 4, 5\}$), and can be annotated as: $(movie\_name, star)$.

When it comes to comments generation task, the two models are both based on Gated Recurrent Unit(GRU). GRU is a variant of Long Short-Term Memory networks(LSTM), and is very similar to LSTM in that GRU also utilizes gates to regulate the information to be kept or discarded at each time step before passing on the long-term and short-term information to the next shell. Both LSTM and GRU can diminish potential gradient explosion or vanishing caused by the flawed structure of Recurrent Neural Network(RNN). However, GRU and LSTM vary in the number of gates and the means they kept internal memory. To be more specific, GRU runs much faster than LSTM. We first present a model which takes topics as inputs and generates a sentence-level text as outputs. Further, the second model has the exact input, output and thus functionality as the first model, except that the second model employs the attention mechanism. The two models are parallel models to indicate the power of attention mechanism and how GRU works at sentence-level text generation.

Notably, we have also adopted adaptive softmax, an approximate strategy for training models with large output spaces. It is most effective when the label distribution is highly imbalanced, for example in natural language modelling, where the word frequency distribution approximately follows the Zipf's law. Adaptive softmax enables our learning curve to converge quicker and our generated texts more reasonable.

3.1 Topic-based GRU Model

Since our comments generation model is directly based upon topics, we input topics straightforward without any further processing. The original GRU model take a word embedding as the input, while our upgraded topic-based GRU model takes a weight-averaged topic and a word as the input. In detail, in our models, all words are mapped into an embedding space with a dimension of 300. Therefore, for each topic $(movie\_name, star)$, a 600 dimensional vector is generated. Along with the input word, which is also a 300 dimensional vector, the input dimension is 900 in total.

Let's take a closer look at the detailed words embeddings. As mentioned above, the value of the corresponding star for each movie name only falls in $\{1, 2, 3, 4, 5\}$, with smaller number indicating

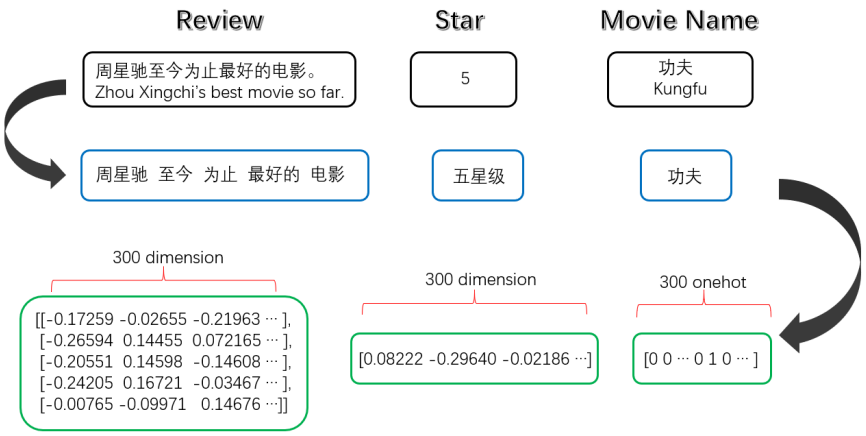

Figure 2: Data Processing Procedure

more negative, less positive comment. Also, the words embeddings for 'one star', 'two stars', etc. is a good representation for its word semantics. On the contrary, for each movie name, it is relatively difficult for tokenizers to convert it into its corresponding word embedding. Even if a word embedding is generated, it is not necessarily a good representation for the word semantics. As a result, we decide to build a 300 dimension one-hot vector for each movie name.

At each time step t of the generation, the prediction for the word y_t is based on the "current" hidden representation h_t, which can be formulated as follows:

$$P(y_t|y_{t-1}, T) = \textit{Ada\_Softmax}(g(h_t))$$

where g(·) is a linear activation function, *Ada_Softmax* is a newly-updated class $AdaptiveLogSoftmaxWithLoss()$ in $torch.nn$ module, T is the input topic embeddings(namely the first 600 dimension of the total 900 input vector).

Meanwhile, before each prediction, h_t is updated by the "candidate" hidden representation $\widetilde{h_t}$ and "previous" hidden representation h_{t-1}, which is:

$$h_t = (1 - z_t) * h_{t-1} + z_t * \widetilde{h_t}$$

and

$$\widetilde{h_t} = tanh(h_{t-1}, T)$$

where z_t is the update gate in GRU.

3.2 Topic-Attention GRU Model

Considering different emphasis of generated words on two aspects of topics, movie name and star respectively, attention mechanism is introduced therefore. The alignment score is the essence of the attention mechanism, as it quantifies the amount of "Attention" the current cell will place on each of the previous outputs when producing the next output. Based upon the previous model, the semantics of the two topic word is converted to the generated words by an attention component, which outputs two scalars:α_1 and α_2, where α_1 represents the score of movie name and α_2 for star.

At each time step t, the topic representation T_t can be formulated as follows:

$$T_t = \alpha_{t1} * movie\_name + \alpha_{t2} * star$$

where T_t is a 300 dimension vector and α_{ti} is derived by:

$$\alpha_{ti} = \frac{exp(g_{ti})}{exp(g_{t1}) + exp(g_{t2})}$$

and

$$g_{ti} = V_a^T tanh(W_a h_{t-1} + U_a topic_i)$$

where $topic_1$ = movie_name and $topic_2$ = star, while V_a, W_a and U_a are three matrices that need to be optimized when training models. Hence, the prediction of the next word h_t can be defined as follow:

$$P(y_t|y_{t-1}, T_t) = \textit{Ada\_Softmax}(g(h_t)))$$

and h_t is still updated as:

$$h_t = (1 - z_t) * h_{t-1} + z_t * \widetilde{h_t}$$

and

$$\widetilde{h}_t = tanh(h_{t-1}, T_t)$$

where tanh$(\cdot)$ is an activation function with parameters determined by the structure of GRU.

Additionally, in our topic-attention model, we have also upgraded the way embedding movie names. After a forementioned one-hot processing, one-hot-processed movie names are sent to a linear layer to be coverted to a 300 dimension vector, which is exactly the dimension of a word embedding in our model. By doing so, we hope to make the model learn what the movie names mean by itself. The parameters of the linear layer is optimized during training to learn the semantics of movie names.

4 Emotions Analysis Models

Emotions Generation Models aim to score each comment with star ranging from one to five. It's easily known that the bigger the score, the more positive the emotion.

Since the Comments Generation Model is very similar to Emotions Analysis Model in essence, we opt to modify the above two Comments Generation Models to meet the demand of our emotion analysis task. Two linear layers for classification of emotions are appended after the hidden state generated by the last word of the comment, which outputs the probability of one star to five stars respectively.

Due to the goal of Emotions Analysis Model, it is obvious that the ground truth of emotions analysis models, to be more detailed, the star of the comment can not serve as inputs. Therefore, we configure the star of each comment into all zeros. As for movie names, it is optional whether each movie name be served as part of the inputs. Thus, **two** separate models are trained depend upon whether the movie name of the comment serves as part of the inputs.

As mentioned before, Comments Generation Models have a lot in common with our present mentioned models–Emotions Analysis Models, and we have pre-trained Comments Generation Models. As a result, the parameters of the shared yet already-trained layers of two large-scale models in Comment Generation Models can be directly transplanted into Emotions Analysis Models. In order to indicate the effectiveness of parameter-optimization in Comments Generation Models, randomly-initialized-parameter Emotions Generation Models are created for **comparison**, both vividly depicted in Fig.3

5 Experiment

We describe experiment settings and report proposed results in this section.

5.1 Experimental Settings

We have crawled 20 movies with Chinese characteristics and for each movie 1500 movie comments are also obtained. We randomly choose 95% of the dataset as out training set, while the remaining 5% serves as testing set. Meanwhile, **beam search** is employed for generation of comments. When the sequence is being constructed, instead of greedily choosing the most likely next step word, the beam search expands all possible next step words and keeps the k most likely, where k is a user-specified parameter and controls the number of beams or parallel searches through the sequence of probabilities. Here, we tune the parameter k to equal to 3.

5.2 Evaluation Methods

Bilingual Evaluation Understudy(a.k.a BLEU), though not flawless, is a widely-applied automatic(thus impersonal) evaluation method of machine translation systems. We import relevant BLEU libraries in our source code and utilize it as the automatic metric for our sentence-level comments generation.

5.3 Experimental Results

In this part, we talk about our experimental results of the Comments Generation Models and Emotions Analysis Models, respectively. Detailed results are shown as follows.

5.3.1 The Result of Comment Generation

We use two topic-based models, as we have mentioned before. The first is very straightforward. Each input contains the composition of topics. The other encodes the topic, and use attention mechanism to get weighted-average topic.

We generate comments for twenty movies which are the exact movies we have crawled, and for each movie there are five stars. A summation of one hundred comments are generated by our comments generation model. We list five comments within the range of one star to five stars in

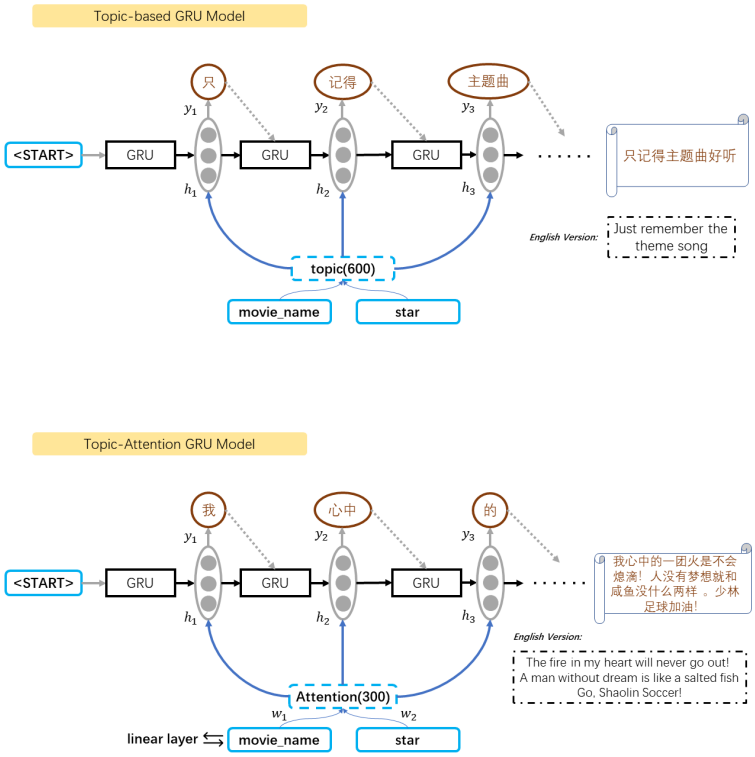

Figure 3: The basic framework for Topic-based GRU Model and Topic-attention GRU Model.

Fig.4 and Fig.5. As we can see from the two pictures, from one star to five stars, the comments are gradually more positive towards the movie. Further, when we compare the result of two movies, it is easy to find that the comments of each movie agree with their movie name, since some scenarios or leading actor or actress are mentioned in the comment.

The result of comparison is somehow counter-intuitive. The first simple model gets better score than the second model, which is based on the attention-mechanism. The results can be seen in Table2. We speculate that the reasons behind may lie in that the second model is hard to train, not only in the forward speed, but also in the learning speed. As a result, we have not made full use of the potential of the model. Besides, the topics we choose aren't very diversely separated, which constrain the functionality of the model based on attention mechanism.

Another fact is also noteworthy. We try to generate comments both with and without movie names. We have already mentioned the comments with movie names as the input. Amazingly, if we

Movie_name	Star	Comments	BLEU
宝莲灯	一星级	这我特么都看过?	1.00
	二星级	只记得三圣母很美	1.00
	三星级	舒畅很可爱。	1.00
	四星级	童年的回忆啊	1.00
	五星级	喜欢曹俊和舒畅，那个时候就爱…	0.81

Figure 4: Five generated comments for Bao Lian Deng from one star to five star, respectively

Movie_name	Star	Comments	BLEU
叶问	一星级	好垃圾的电影，操蛋	1.00
	二星级	甄子丹的演技一如既往地烂，但是武打演的很好	0.90
	三星级	甄子丹的功夫华丽的很。	1.00
	四星级	甄子丹的一次让我惊艳到~	0.35
	五星级	看多看后觉得很好看，比一如既往的人，还是也很喜欢很给力	0.01

Figure 5: Five generated comments for Ye Wen from one star to five star, respectively

do not input the movie names, the generated comments match the input stars very well. For example, when the input is five-star, the comments generated are like that "I love the movie deeply" or "It's a great movie" or "The movie desires five-star." which shows that it can indeed learn huge amounts of very generalized parameters.

5.3.2 The Result of Classification

Since the hidden state of the above model we have trained is good at encoding comments, we

Table 2: The BLEU Score of Each Model

model	BLEU score
Topic-based GRU	0.81
Topic-weighted GRU	0.45

are able to slightly alter the above model to fit into our second classification model, by simply changing the final layer. In order to extend our study of comments' emotions, we make comparisons between two methods. The first inherits the parameters of the above model, while the other initializes parameters' weights from scratch. As far as we are concerned, what we expect to see is that in terms of the learning speed, the model with loaded parameters will be much faster than the other while in the aspect of accuracy, the performance of the model with inherited parameters should be equal or better than the other.

Then it comes the evaluation of our second model. At first, we split 95% data as training set, and the rest as test set. Only if the predicted star matches the ground truth, we think of it correct. Otherwise, it's incorrect. However, the result is very bad, since the accuracy is only a little bigger than 40%. Therefore, we have to change the evaluation method. We choose one star as the negative comment, two to four stars as the middle comment, and five stars as the positive comment. Then, the accuracy is a little larger than 70%. However, we are still not satisfied with our result. We think the reason may be that the test set is too small. Then we readjusted the data set. We split 60% data into training set and the remaining 40% into test set. The accuracy in test set achieves 89.51% when classified in five categories and 95.87% when classified in three categories.This is really inspiring.

The visualization of the result of the learning curve is shown in Fig.6. The accuracy of comments' classification in five categories can be seen in Fig.7 while the three categories can be seen in Figure.8. In terms of learning speed, The results are within our expectation. The learning speed of the model with loaded or inherited parameters is far faster. However, in terms of the accuracy, the adaptation of parameters learned from scratch is better than parameters inherited from the previous model. It can be naturally deducted that the hidden state of above model doesn't completely adapt to our new task. Therefore, the learned parameters, though speed up our learning speed at the beginning, hinder the final performance of the second model.

Furthermore, We are wondering whether the model can understand the emotions if it know other topic. The guess we make is that other known topic does help improve the performance. However, from Figure 6, Figure 7 and Figure 8, we can easily find that if the model is trained from inheriting parameters, new input topic help a little. If the model is train from random parameters, extra topic input will do harm to the result. The reason is easily to find. Because the model inheriting parameters have already leaned the topic, with topic input, the hidden state will have a good meaning. However, if the model trained from random parameters, the topic doesn't have a good correlation with emotion, as consequence of which, the topic input will inhibit learning as the extra and useless information.

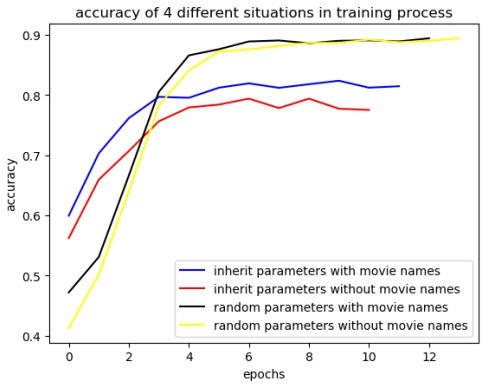

Figure 6: The accuracy of test set in comparison with whether to load parameters

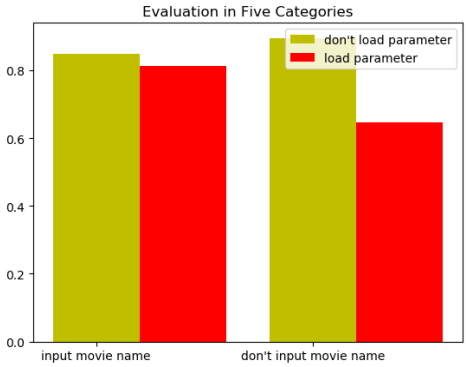

Figure 7: The result of evaluation in five categories

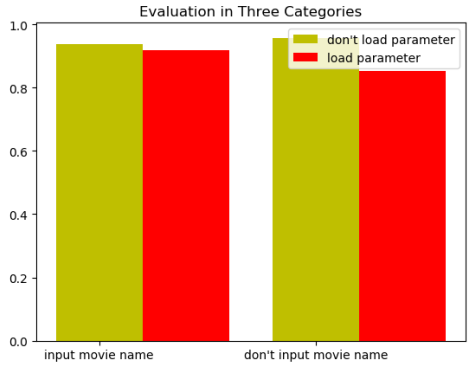

Figure 8: The result of evaluation in three categories

6 Related Work

In this section, we introduce some related works concerning similar tasks and researching directions with our project.

NLG is a quite popular researching field in recent years concerning NLP. Traditional text generation methods based on rules(Duma D, 2013) and templates(Zhou Qingyu, 2017) have fixed text format and lack semantic information. As a result, these methods failed to produce texts with rich contents and diversified styles. With the dramatic development of deep learning methods, many text generation methods based on deep neural networks are proposed. RNN and LSTM are being employed widely(A, 2013 08 04), as can the text generated by deep neural networks does not understand the exact topic and the text itself is not fluent at all sometimes. Again, some solutions have already been raised by many talents. For example, add topic words to text evidently(Ghazvininejad M, 2016), select words that are similar to topic words and reorganize them all into new texts(Wang Zhe, 2016), and take into account the attention mechanism in RNN(Feng Xiaocheng, 2018) and (Xing Chen, 2017). These methods do relieve the aforementioned two critical problems, but they do not solve the following-up problems, such as solo theme, subjects out of control, and the issue of the coverage of topic words.

In our model, we employ GRU with multiple advanced techniques: attention mechanism, beam search and adaptive softmax layer to aid in generating comments. As discussed in the previous section, our generated comments agree with the input movie name and star.

7 Conclusion and Future Work

In conclusion, the Comments Generating Model which is directly based on topic and can generate comments of high quality than the other. This means that we can input different topics, and obtain the corresponding comments, which can save our time to some degree. As for our second model, we train an emotion analysis model using transfer learning, which can achieve a very high accuracy in test set.It means we can input a comment ,and get the star of it with very high accuracy in a fairly quick manner. Here, although the second model which does not inherit the first model's parameters performs better than that that whit the inherits, the difference is not obvious. However, the one that inherits is far more efficient in training and testing. To balance the accuracy and efficiency, we prefer the one that inherits the parameters. At last, our group successfully reach the demand of the project. In the meantime, we learn a way to make machine generate sentences just like humanbeings as well as evaluate the emotions of the reviews on the website.

In the future, we want to upgrade our networks by integrating more logical knowledge and reasonable common sense and raise its error-tolerant rate. For extension, we plan to outstretch our models to other forms of literature genres, such as: tedious book reviews. Our model should also be tolerant with other popular languages as well, especially English. Besides, we really need to expand the size of our Word2vec dictionary and our model may be capable of recognize and even generate emojis and other semantic elements in order to behave more like a human reviewer. In case of illegal use of this kind of techniques to deliberately generate machine reviews for certain purposes, a supervisory mechanism for reviewing website should be built up in future.

Acknowledgements

This work is completed by Ding Pan, Bo Yue and JiaSheng Xu, and is supported by Prof. Tu and teaching assistant, all from Shanghai Jiaotong University(SJTU).

References

Graves A. 2013-08-04. Generating sequences with recurrent neural networks. *ArXiv: Neural and Evolutionary Computing*.

Klein E. Duma D. 2013. Generating natural language from linked data: unsupervised template extraction. *Proc of the 10th International Conference on Computational Semantics. Stroudsburg.*, pages 83–94.

Liu Jiahao et al. Feng Xiaocheng, Liu Ming. 2018. Topic-to-essay generation with neural networks. *Proc of the 27th International Joint Conference on Artificial Intelligence and the 23rd European Conference on Artificial Intelligence. San Francisco*, pages 4078–4084.

Choi Y et al Ghazvininejad M, Shi Xing. 2016. Generating topical poetry. *Proc of the 2016 Conference on Empirical Methods in Natural Language Processing. Stroudsburg,*, pages 1183–1191.

Wu Haiyang et al. Wang Zhe, He Wei. 2016. Chinese poetry generation with planning based neural network. *Proc of the 26th International Conference on Computational Linguistics.*, pages 1051–1060.

Wu Yu et al Xing Chen, Wu Wei. 2017. Topic aware neural response generation. *Proc of the 31th AAAI Conference on Artificial Intelligence. Menlo Park*, pages 3351–3357.

Wei Furu et al. Zhou Qingyu, Yang Nan. 2017. Selective encoding for abstractive sentence summarization. *Proc of the 55th Annual Meeting of the Association for Computational Linguistics. Stroudsburg.*, pages 1095–1104.

Group Work Division

In this section, our group's work division is listed, as demanded by the course project requirements.

Ding Pan Raise the core idea; Preprocess the data; Build the prototype models by himself; Design the main framework; Write this paper.

Bo Yue Participate in establishing research ideas and topics of this paper; Take part in building and debugging the two models; Build metrics for evaluation; Make presentation at class; Write the project proposal; Write the major part of this paper.

Jiasheng Xu Help raise this topic; Crawl the movie reviews from Douban website and process the data; Take part in building and debugging the two models; Take care of as well as run all the code, attain the results and visualize them; Write the mid-term report; Write this paper.