Towards Automatic Generation of Product Reviews from Aspect-Sentiment Scores

Hongyu Zang and **Xiaojun Wan**
Institute of Computer Science and Technology, Peking University
The MOE Key Laboratory of Computational Linguistics, Peking University
{zanghy, wanxiaojun}@pku.edu.cn

Abstract

Data-to-text generation is very essential and important in machine writing applications. The recent deep learning models, like Recurrent Neural Networks (RNNs), have shown a bright future for relevant text generation tasks. However, rare work has been done for automatic generation of long reviews from user opinions. In this paper, we introduce a deep neural network model to generate long Chinese reviews from aspect-sentiment scores representing users' opinions. We conduct our study within the framework of encoder-decoder networks, and we propose a hierarchical structure with aligned attention in the Long-Short Term Memory (LSTM) decoder. Experiments show that our model outperforms retrieval based baseline methods, and also beats the sequential generation models in qualitative evaluations.

1 Introduction

Text generation is a central task in the NLP field. The progress achieved in text generation will help a lot in building strong artificial intelligence (AI) that can comprehend and compose human languages.

Review generation is an interesting subtask of data-to-text generation. With more and more online trades, it usually happens that customers are lazy to do brainstorming to write reviews, and sellers want to benefit from good reviews. As we can see, review generation can be really useful and worthy of study. But recent researches on text generation mainly focus on generation of weather reports, financial news, sports news (Konstas, 2014; Kim et al., 2016; Zhang et al., 2016), and so on. The task of review generation still needs to be further explored.

Think about how we generate review texts: we usually have the sentiment polarities with respect to product aspects before we speak or write. Inspired by this, we focus on study of review generation from structured data, which consist of aspect-sentiment scores.

Traditional generation models are mainly based on rules. It is time consuming to handcraft rules. Thanks to the quick development of neural networks and deep learning, text generation has achieved a breakthrough in recent years in many domains, e.g., image-to-text (Karpathy and Fei-Fei, 2015; Xu et al., 2015), video-to-text (Yu et al., 2016), and text-to-text (Sutskever et al., 2014; Li et al., 2015), etc. More and more works show that generation models with neural networks can generate meaningful and grammatical texts (Bahdanau et al., 2015; Sutskever et al., 2011). However, recent studies of text generation mainly focus on generating short texts of sentence level. There are still challenges for modern sequential generation models to handle long texts. And yet there is very few work having been done in generating long reviews.

In this paper, we aim to address the challenging task of long review generation within the encoder-decoder neural network framework. Based on the encoder-decoder framework, we investigate different models to generate review texts. Among these models, the encoders are typically Multi-Layer Perceptron (MLP) to embed the input aspect-sentiment scores. The decoders are RNNs with LSTM units, but differ in architectures. We proposed a hierarchi-

Proceedings of The 10th International Natural Language Generation conference, pages 168–177,
Santiago de Compostela, Spain, September 4-7 2017. ©2017 Association for Computational Linguistics

cal generation model with a new attention mechanism, which shows better results compared to other models in both automatic and manual evaluations based on a real Chinese review dataset.

To the best of our knowledge, our work is the first attempt to generate long review texts from aspect-sentiment scores with neural network models. Experiments proved that it is feasible to general long product reviews with our model.

2 Problem Definition and Corpus

To have a better understanding of the task investigated in this study, we'd like to introduce the corpus first.

Without loss of generality, we use Chinese car reviews in this study and reviews in other domains can be processed and generated in the same way. The Chinese car reviews are crawled from the website AutoHome[1]. Each review text contains eight sentences describing eight aspects[2], respectively: 空间*/Space*, 动力*/Power*, 控制*/Control*, 油耗*/Fuel Consumption*, 舒适度*/Comfort*, 外观*/Appearance*, 内饰*/Interior*, and 性价比*/Price*. Each review text corresponds to these eight aspects and the corresponding sentiment ratings, and the review sentences are aligned with the aspects and ratings. So we may split the whole review into eight sentences when we need. Note that the sentences in each review are correlated with each other, so if we regard them as independent sentences with respect to individual aspect-sentiment scores, they probably seem pretty mendacious when put altogether. We should keep each review text as a whole and generate the long and complete review at one time, rather than generating each review sentence independently. Specifically, we define our task as generating long Chinese car reviews from eight aspect-sentiment scores.

The raw data are badly formatted. In order to clean the data, we keep the reviews whose sentences corresponding to all the eight aspects. And we skip the reviews whose sentences are too long or too short. We accept length of 10 to 40 words per sentence. We use Jieba[3] for Chinese word segmentation. Note that each review text contains eight sentences, where each sentence has 24 Chinese characters on average. The review texts in our corpus are actually very long, about 195 Chinese characters per review.

The rating score for each aspect is in a range of [1, 5], and we regard rating 3 as neutral, and normalize ratings into [-1.0, 1.0] by Equation (1)[4], and the sign of a normalized rating means the sentiment polarity. For instance, if the original ratings for all eight aspects are [1,2,3,4,5,4,3,2], we will normalize it into[-1.0,-0.5,0.0,0.5,1.0,0.5, 0.0,-0.5] and use the normalized vector as the input for review generation.

$$x' = \frac{x - \frac{Max+Min}{2}}{\frac{Max-Min}{2}} \tag{1}$$

And finally, we get 43060 pairs of aspect-sentiment vectors and the corresponding review texts, in which there are 8340 different inputs[5]. Then we split the data randomly into training set and test set. The training set contains 32195 pairs (about 75%) and 6290 different inputs, while the test set contains the rest 10865 pairs with 2050 different inputs. The test set does not overlap with the train set with respect to the input aspect-sentiment vector.

Furthermore, we transform the input vector into aspect-oriented vectors as input for our models. For each aspect, we use an additional one-hot vector to represent the aspect, and then append the one-hot vector to the input vector. For example, if we are dealing with a specific aspect *Power* corresponding to a one-hot vector [0,1,0,0,0,0,0,0] for the above review with input vector [-1.0,-0.5,0.0,0.5,1.0,0.5,0.0,-0.5], the new input vector with respect to this aspect is actually [-1.0,-0.5,0.0,0.5,1.0,0.5,0.0,-0.5,0,1,0,0,0,0,0,0]. Each new input vector is aligned with a review sentence. Similarly, we can get eight new vectors with respect to the eight aspects as input for our models.

[1] www.autohome.com.cn

[2] In fact, there may be multiple grammatical sentences describing one single aspect. But for simplification, we define the sequence of characters describing the same aspects as a sequence.

[3] github.com/fxsjy/jieba

[4] We set the origin rating as x, and the normalized rating as x'. Max and Min is the maximum and minimum value out of all the original ratings in the dataset, or rather, 5 and 1.

[5] We allow multiple gold-standard answers to one input.

3 Preliminaries

In this section, we will give a brief introduction to LSTM Network (Hochreiter and Schmidhuber, 1997).

3.1 RNN

RNN has been widely used for sequence generation tasks (Graves, 2012a; Schuster and Paliwal, 1997). RNN accepts sequence of inputs $X = \{x_1, x_2, x_3, ..., x_{|X|}\}$, and gets h_t at time t according to Equation (2).

$$h_t = W_H \times \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \tag{2}$$

3.2 LSTM Network

An LSTM network contains LSTM units in RNN and an LSTM unit is a recurrent network unit that excels at remembering values for either long or short durations of time(Graves, 2012b; Sundermeyer et al., 2012). It contains an input gate, a forget gate, an output gate and a memory cell. Respectively, at time t, we set the above parts as i_t, f_t, o_t, c_t. In an LSTM network, we propagate as Equation (3)(4)(5).

$$\begin{bmatrix} i_t \\ f_t \\ o_t \end{bmatrix} = sigmoid\left(\begin{bmatrix} W_I \\ W_F \\ W_O \end{bmatrix} \times \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \right) \tag{3}$$

$$c_t = i_t \times tanh\left(W_C \times \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \right) + f_t \times c_{t-1} \tag{4}$$

$$h_t = c_t \times o_t \tag{5}$$

In the past few years, many generation models based on LSTM networks have given promising results in different domains (Xu et al., 2015; Shang et al., 2015; Wu et al., 2016). Compared to other network units of RNN, like GRU (Chung et al., 2014), LSTM is considered the best one in most cases.

4 Review Generation Models

4.1 Notations

We define our task as receiving a vector of aspect-sentiment scores V_s to generate review texts, which is a long sequence of words $Y\{y_1, y_2, \ldots, y_{|Y|-1}, \langle EOS \rangle\}$ ($\langle EOS \rangle$ is the special word representing the end of a sequence). As mentioned in section 2, we also transform an input vector V_s into a series of new input vectors $\{V_1, V_2, \ldots, V_8\}$ with respect to eight aspects for our models. More specifically, in order to obtain each V_i, we append a one-hot vector representing a specific aspect to V_s. That is, $V_i = [V_s, O]$, where O is a one-hot vector with the size of eight, and only the ith element of O is 1.

We have three different kinds of embeddings: E^W stands for word embedding, E^V stands for embedding of the input vector by a MLP encoder, and E^C stands for embedding of context sentences. There will be subscripts specifying the word, the vector, and the context.

And in LSTM, h is a hidden vector, x is an input vector, P is the possibility distribution, y' is the predicted word, and t is the time step.

4.2 Sequential Review Generation Models (SRGMs)

SRGMs are similar to the popular Seq2Seq models (Chung et al., 2014; Sutskever et al., 2011), except that it receives inputs of structured data (like aspect-sentiment scores) and encodes them with an MLP.

The encoder's output E^V_s is treated as the initial hidden state h_0 of the decoder. And the initial input vector is set as the word embedding of $\langle BOS \rangle$ ($\langle BOS \rangle$ is the special word representing the begin of a sequence). Then the decoder proceeds as a standard LSTM network.

At time $t(t \geq 1)$, the hidden state of the decoder h_t is used to predict the distribution of words by a softmax layer. We will choose the word with max possibility as the word predicted at time t, and the word will be used as the input of the decoder at time $t + 1$.

This procedure can be formulated as follows:

$$h_0 = E^V_s = MLP(V_s) \tag{6}$$

$$x_1 = E^W_{\langle BOS \rangle} \tag{7}$$

$$h_t = LSTM(h_{t-1}, x_t) \tag{8}$$

$$P_t = softmax(h_t) \tag{9}$$

$$y'_t = argmax_w(P_{t,w}) \tag{10}$$

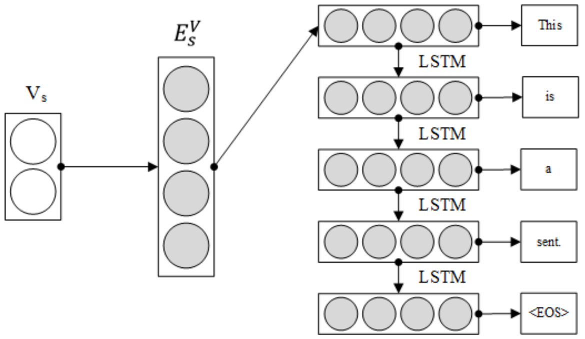

Figure 1: The architecture of SRGM-w.

$$x_{t+1} = E^W_{y'_t} \tag{11}$$

In each training step, we adopt the negative likelihood loss function.

$$Loss = -\frac{1}{|Y|}\sum_t logP_{t,y_t} \tag{12}$$

However, Sutskever et al. (2014) and Pouget-Abadie et al. (2014) have shown that standard LSTM decoder does not perform well in generating long sequences. Therefore, besides treating the review as a whole sequence, we also tried splitting the reviews into sentences, generating the sentences separately, and then concatenating the generated sentences altogether. Respectively, we name the sequential model generating the whole review as SRGM-w, and the one generating separate sentences as SRGM-s.

4.3 Hierarchical Review Generation Models (HRGMs)

Inspired by Li et al. (2015), we build a hierarchical LSTM decoder based on the SRGMs. Note that we have two different LSTM units in hierarchical models, in which the superscript S denotes the sentence−level LSTM, and the superscript P denotes the paragraph−level one. And t is the time step notation in the sentence decoder, while T is the time step notation in the paragraph decoder. Both the time step symbols are put in the position of subscripts.

There is a one-hidden-layer-MLP to encode the input vector into E_s^V. $LSTM^P$ receives E_s^V as the initial hidden state, and the initial input x_1^P is a zero vector. At time $T(T \geq 1)$, the output of $LSTM^P$ is used as the initial hidden state of $LSTM^S$. And then $LSTM^S$ works just like the LSTM decoder in SRGMs. The final output of $LSTM^S$ is treated as the embedding of the context sentences E_T^C, which is also the input of $LSTM^P$ at time $T+1$. We call this hierarchical model HRGM-o.

$$h_0^P = E_s^V = MLP(V_s) \tag{13}$$

$$x_1^P = \mathbf{0} \tag{14}$$

$$h_T^P = LSTM^P(h_{T-1}^P, x_T^P) \tag{15}$$

$$h_{T,0}^S = h_T^P \tag{16}$$

$$h_{T,t}^S = LSTM^S(h_{T,t-1}^S, x_{T,t}^S) \tag{17}$$

$$P_{T,t} = softmax(h_{T,t}^S) \tag{18}$$

$$y'_{T,t} = argmax_w(P_{T,t,w}) \tag{19}$$

$$x_{T,t+1}^S = E^W_{y'_{T,t}} \tag{20}$$

$$x_{T+1}^P = E_T^C = h^S_{T,|Y_T|} \tag{21}$$

In the experiment results of HRGM-o, we find that the model has its drawback. In some test cases, the output texts miss some important parts of the input aspects.

As many previous studies have shown that the attention mechanism promises a better result by considering the context (Bahdanau et al., 2015; Fang et al., 2016; Li et al., 2015). We adopt attention to the generation of each sentence, which is aligned to the sentence's main aspect.

Different from the attention mechanism mentioned in previous studies, in our situation, we have the alignment relationships between aspect-sentiment ratings and sentences, which are natural attentions to be used in the generation process. By applying additional input vector V_T at each time step T, we obtain the initial hidden state of $LSTM^S$ from two source vectors E_T^V and h_T^P. Therefore, we simply train a gate vector g to control the two parts of information. The encoding of V_T is similar to Equation (13), but with different parameters. In brief, we change Equation (16) to Equation (22)(23).

$$E_T^V = MLP'(V_T) \tag{22}$$

$$h_{T,0}^S = \begin{bmatrix} h_T^P \\ E_T^V \end{bmatrix} \times [g, \mathbf{1} - g] \tag{23}$$

Based on all of these, we propose a hierarchical model with a special aligned attention mechanism as shown in Figure 2. We call the model HRGM-a.

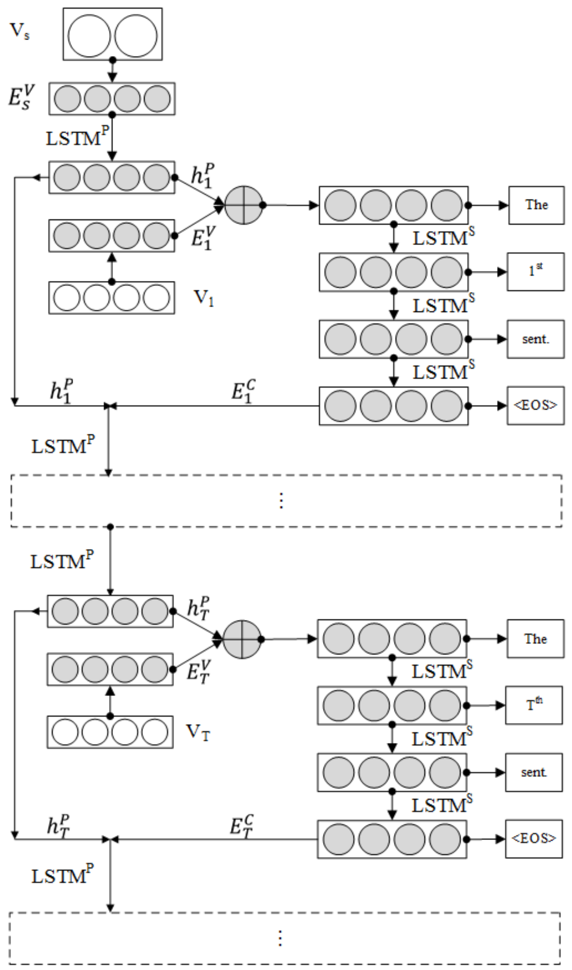

Figure 2: The architecture of HRGM-a.

5 Experiments

5.1 Training Detail

We implemented our models with TensorFlow 1.10[6], and trained them on an NVIDIA TITANX GPU (12G).

Because the limitation of our hardware, we only do experiments with one layer of encoder and one layer of LSTM network. The batch size is 4 in HRGMs, and 32 in SRGMs. The initial learning rate is set to 0.5, and we dynamically adjust the learning rate according to the loss value. As experiments show that the size of hidden layer does not affect the results regularly, we set all of them to 500.

All the rest parameters in our model can be learned during training.

5.2 Baselines

Apart from SRGM-w and SRGM-s, we also developed several baselines for comparison.

- **Rand-w**: It randomly chooses a whole review from the training set.
- **Rand-s**: It randomly choose a sentence for each aspect from the training set and concatenates the sentences to form a review.
- **Cos**: It finds a sentiment vector from the training set which has the the largest cosine similarity value with the input vector, and then returns the corresponding review text.
- **Match**: It finds a sentiment vector from the training set which has the maximum number of rating scores matching exactly with that in the input vector, and then returns the corresponding review text.
- **Pick**: It finds one sentence for each aspect respectively in the training set by matching the same sentiment rating, and then concatenates them to form a review.

Generally speaking, models in this paper are divided into four classes. The first class is lower bound methods (Rand-w, Rand-s), where we choose something from the training set randomly. The second one is based on retrieval (Cos, Match, Pick), and we use similarity to decide which to choose. The third one is sequential generation models based on RNNs (SRGM-w, SRGM-s). And the last one is hierarchical RNN models to handle the whole review generation (HRGM-o, HRGM-a).

5.3 Automatic Evaluation

We used the popular BLEU (Papineni et al., 2002) scores as evaluation metrics and BLEU has shown good consistent with human evaluation in many machine translation and text generation tasks. High BLEU score means many n-grams in the hypothesis texts meets the gold-standard references. Here, we report BLEU-2 to BLEU-4 scores, and the evaluation is conducted after Chinese word segmentation.

The only parameters in BLEU is the weights W for n-gram precisions. In this study, we set W as average weights ($W_i = \frac{1}{n}$ for BLEU-n evaluation). As for multiple answers to the same input, we put all of them into the reference set of the input.

The results are shown in Table 1. Retrieval based

[6] github.com/tensorflow/tensorflow/tree/r0.10

	BLEU-2	**BLEU-3**	**BLEU-4**
Rand-w	0.1307	0.0378	0.0117
Rand-s	0.1406	0.0412	0.0124
Cos	0.1342	0.0403	0.0129
Match	0.1358	0.0423	0.0136
Pick	0.1427	0.0434	0.0133
SRGM-w	0.1554	0.0713	0.0307
SRGM-s	0.1709	0.0829	0.0369
HRGM-o	0.1850	0.0854	0.0334
HRGM-a	**0.1985**	**0.0942**	**0.0412**

Table 1: The results of BLEU evaluations.

baselines get low BLEU scores in BLEU-2, BLEU-3 and BLEU-4. Among these models, Cos and Match even get lower BLEU scores than the lower bound methods in some BLEU evaluations, which may be attributed to the sparsity of the data in the training set. Pick is better than lower bound methods in all of the BLEU evaluations. Compared to the retrieval based baselines, SRGMs get higher scores in BLEU-2, BLEU-3, and BLEU-4. It is very promising that HRGMs get the highest BLEU scores in all evaluations, which demonstrates the effectiveness of the hierarchical structures. Moreover, HRGM-a achieves better scores than HRGM-o, which verifies the helpfulness of our proposed new attention mechanism.

In all, the retrieval models and sequential generation models can not handle long sequences well, but hierarchical models can handle long sequences. The reviews generated by our models are of better quality according to BLEU evaluations.

5.4 Human Evaluation

We also perform human evaluation to further compare these models. Human evaluation requires human judges to read all the results and give judgments with respect to different aspects of quality.

We randomly choose 50 different inputs in the test set. For each input, we compare the best models in each class, specifically, Rand-s, Pick, SRGM-s, HRGM-a, and the Gold (gold-standard) answer. We employ three subjects (excluding the authors of this paper) who have good knowledge in the domain of car reviews to evaluate the outputs of the models. The outputs are shuffled before shown to subjects. Without any idea which output belongs to which model, the subjects are required to rate on a 5-pt Likert scale[7] about readability, accuracy, and usefulness. In our 5-pt Likert scale, 5-point means “very satisfying”, while 1-point means “very terrible”. The ratings with respect to each aspect of quality are then averaged across the three subjects and the 50 inputs.

To be more specific, we define readability, accuracy, and usefulness as follows. Readability is the metric concerned with the fluency and coherence of the texts. Accuracy indicates how well the review text matches the given aspects and sentiment ratings. Usefulness is more subjective, and subjects need to decide whether to accept it or not when the text is shown to them. The readability, accuracy, even the length of the review text will have an effect on the usefulness metric.

	Readability	**Accuracy**	**Usefulness**
Gold	4.61	4.41	4.39
Rand-s	4.44	3.21	3.52
Pick	**4.55**	4.15	4.20
SRGM-s	4.51	4.21	4.21
HRGM-a	4.52	**4.33**	**4.26**

Table 2: Human evaluation results of typical models. We set the best result of each metric in bold except for Gold-Standard.

The results are shown in Table 2. We can see that in human evaluations, all the models get high scores in readability. The readability score of our model HRGM-a is very close to the highest readability score achieved by Pick. Rand-s gets the worst scores for accuracy and usefulness, while the rest models perform much better in these metrics. Compared to the strong baselines Pick and SRGM-s, although our model is not the best in readability, it performs better in accuracy and usefulness. The results also demonstrate the efficacy of our proposed models.

5.5 Samples

To get a clearer view of what we have done and have an intuitive judgment of the generated texts, we present some samples in Table 3.

In Table 3, the first three samples are output texts of Gold-Standard, Pick, and our model HRGM-a for the same input. And in the last sample, we

[7] en.wikipedia.org/wiki/Likert_scale

Inputs	Outputs
Gold-Standard Space: 3 Power: 4 Control: 5 Fuel: 3 Comfort: 3 Appearance: 5 Interior: 4 Price: 4	后备箱的空间还是蛮大的，就是后排的空间比较小，座椅也不平整。动力还行吧，只要舍得给油，还说的过去。方向盘精准度高，路况反应清晰。可能是因为轮胎薄的原因吧，自动档的油耗有点高，市区油耗在10个左右，高速最多7个油。座椅还是蛮舒适的，就是行车中噪音比较大，建议做个全车隔音比较好。小$\langle UNK\rangle$ 的颜值在同级别里算高的了，这点比较不错，特别是那个战斧轮毂。用料还行，偶尔会有点小异响，这个价位的车差不多$\langle UNK\rangle$这样。不错，对得起这个价了，毕竟价钱摆在那里。 Translation: Trunk space is quite large, but the rear space is relatively small, and the seat is not smooth. Power is also okay, as long as willing to give oil. Steering wheel has high precision. Probably because of the reasons for thin tires, fuel consumption of automatic transmission is a bit high, urban fuel consumption in 10 or so, while on high way up to 7 oil. The seat is still quite comfortable, but there is large noise when driving. I propose the car to have a better sound insulation. Little $\langle UNK\rangle$'s appearance is better than others in the same class, and this is quite good. I especially like the Tomahawk wheels. Materials are okay, there is occasionally a little abnormal sound. The price of the car is acceptable. Yeah, worthy of the price $\langle UNK\rangle$. After all, the price is not that high.
Pick Space: 3 Power: 4 Control: 5 Fuel: 3 Comfort: 3 Appearance: 5 Interior: 4 Price: 4	略显狭小，如果坐4个人就很拥挤了。1.4T双增压带来的直观动力表现是不错的，不输给家里的锐志，毕竟1.4的车。新款也把双增压换成单增压，感觉有点不厚道。指哪打哪，宝马的公路操控，应该是比较不错了，很精准，无虚伪，路感强，现在400公里纯市内全程空调12个多，没平均速度可看，2.5的排量这个油耗能够接受。由于空间较大，所以随之舒适性也加了不少分。这个不多说OK！！！满意！！！一般吧，希望有些小细节要改进一下，多听听大家的意见。优惠完之后，觉得性价比还好，要是优惠$\langle UNK\rangle$就不考虑宝来了 Translation: The space is slightly narrow, not enough for four people. The intuitional performance given by 1.4T double booster is good, better the Reiz of mine. After all, 1.4T. The new dual-pressure supercharger is replaced by a single one, where I feel a little unkind. The car goes anywhere I want. BMW's road control should be relatively good. Very accurate, no hypocrisy, strong sense of the road. I have driven the car 400 km along with air condition, the fuel consumption is about 12. I didn't pay attention to the average speed, but I think a displacement of 2.5 is acceptable to have such fuel consumption. Because of the large space, it's very comfortable in the car. I'm very satisfactory with the appearance! About interior, I think there are some small details to improve. The price is good with discount, if not for the discount $\langle UNK\rangle$, I won't consider Bora.
HRGM-a Space: 3 Power: 4 Control: 5 Fuel: 3 **Comfort: 3** Appearance: 5 Interior: 4 Price: 4	空间一般，后排空间有点小。动力方面，提速很快。超车很轻松。操控很好，指向精准，方向盘很轻，高速容易上手。油耗在市区内比较高，高速在7个左右，市区的话油耗在10个左右。**舒适性一般，毕竟是运动型的车**。外观很满意，我喜欢。内饰做工还可以，就是中控的塑料感太强了。性价比很高，这个价位的车，这个配置的值价位，这个价格。性价比配置，这个价位。值！ Translation: The space is just so so, as the rear space is a little small. As for power, it can speed up very quickly, which makes it pretty easy to overtake. The control is good. It's very precise. And the steering wheel is very light, easy to use on highway. Fuel consumption in the urban area is relatively high, about 7 on highway, about 10 in urban roads. **It's not comfortable enough in the car. After all, it is a sports car.** The appearance is very satisfactory. I like it very much. Interiors are ok. But there is too much plastic in center control area. The price/performance ratio is very high. A car at this price, with these configurations, worths buying.
HRGM-a Space: 3 Power: 4 Control: 5 Fuel: 3 **Comfort: 5** Appearance: 5 Interior: 4 Price: 4	空间一般，后排空间有点小，后备箱空间也不错，就是后排座椅不能放倒。动力还不错，提速很快。操控很好，指向精准。油耗还可以，毕竟是2.0的排量，油耗也不高，毕竟是2.0的排量，也不可能我个人开车的原因。**舒适性很好，座椅的包裹性很好，坐着很舒服。**外观很满意，就是喜欢。很有个性。内饰做工一般，但是用料还是很好的，不过这个价位的车也就这样了！性价比不错，值得购买。 Translation: The space is just so so, as the rear space is a little small. The trunk space is also good, but the rear seat cannot be tipped. Power is also OK. The car can speed up very quickly. Control is very good. It goes wherever you want. Fuel consumption is acceptable. After all, with a 2.0 displacement, fuel consumption is not that high. But it can't be my problem. **It's comfortable in the car. The seats are well wrapped, which makes them really comfortable.** The appearance is very satisfactory. I just like the cool features. Interiors are ok. The materials are ok. After all, you can't want more from cars at this price. It's worth buying the car, and I can say that the price/performance ratio is pretty good.

Table 3: Sample reviews. Given the same input, our model can generate long reviews that matches the input aspects and sentiments better than the baseline methods. When we change the input rating for *Comfortable* from middle (3) to high (5), our model can also detect the difference and change the outputs accordingly.

change one rating in the input to show how our model changes the output according to the slight difference in the input.

As we can see, Pick is a little better than our model HRGM-a in text length and content abundance. But the output of Pick has a few problems. For example, there is a serious logic problem in the reviews of *Space* and *Comfort*. It says the car is narrow in *Space*, but the car has a large space in *Comfort*, which violates the context consistency.

What's more, it gives improper review to *Comfort*. Although *Comfort* gets 3-point, the review sentence is kind of positive. And that can be considered as a mismatch with the input. On the contrary, our model produces review texts as a whole and the texts are aligned with the input aspect-sentiment scores more appropriately. All 3-point aspects get neutral or slightly negative reviews, while all 5-point aspects get definitely positive comments. And 4-point aspects also get reviews biased towards being positive.

As for the last example after changing the rating of *Comfort* from 3-point to 5-point, we can see that except for the review sentence for *Comfort*, other sentences do not change apparently. But the review sentence of *Comfort* changes significantly from neutral to positive, which shows the power of our model.

6 Related Work

Several previous studies have attempted for review generation (Tang et al., 2016; Lipton et al., 2015; Dong et al., 2017) . They generate personalized reviews according to an overall rating. But they do not consider the product aspects and whether each generated sentence is produced as the user requires. The models they proposed are very similar to SRGMs. And the length of reviews texts are not as long as ours. Therefore, our work can be regarded as a significant improvement of their researches.

Many researches of text generation are also closely related to our work. Traditional way for text generation (Genest and Lapalme, 2012; Yan et al., 2011) mainly focus on grammars, templates, and so on. But it is usually complicated to make every part of the system work and cooperate perfectly following the traditional techniques, while end-to-end generation systems nowadays, like the ones within encoder-decoder framework (Cho et al., 2014; Sordoni et al., 2015), have distinct architectures and achieve promising performances.

Moreover, the recent researches on hierarchical structure help a lot with the improvement of the generation systems. Li et al. (2015) experimented on LSTM autoencoders to show the power of the hierarchical structured LSTM networks to encode and decode long texts. And recent studies have succefully generated Chinese peotries(Yi et al., 2016) and Song iambics(Wang et al., 2016) with hierarchical RNNs.

The attention mechanism originated from the area of image (Mnih et al., 2014), but is widely used in all kinds of generation models in NLP (Bahdanau et al., 2015; Fang et al., 2016). Besides, attention today is not totally the same with the original ones. It's more a thinking than an algorithm. Various changes can be made to construct a better model.

7 Conclusion and Future Work

In this paper, we design end-to-end models to challenge the automatic review generation task. Retrieval based methods have problems generating texts consistent with input aspect-sentiment scores, while RNNs cannot deal well with long texts. To overcome these obstacles, we proposed models and find that our model with hierarchical structure and aligned attention can produce long reviews with high quality, which outperforms the baseline methods.

However, we can notice that there are still some problems in the texts generated by our models. In some generated texts, the contents are not rich enough compared to human-written reviews, which may be improved by applying diversity decoding methods (Vijayakumar et al., 2016; Li et al., 2016). And there are a few logical problems in some generated texts, which may be improved by generative adversarial nets (Goodfellow et al., 2014) or reinforcement learning (Sutton and Barto, 1998).

In future work, we will apply our proposed models to text generation in other domains. As mentioned earlier, our models can be easily adapted for other data-to-text generation tasks, if the alignment between structured data and texts can be provided. We hope our work will not only be an exploration of review generation, but also make contributions to general data-to-text generation.

Acknowledgments

This work was supported by 863 Program of China (2015AA015403), NSFC (61331011), and Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology). We thank the anonymous reviewers for helpful comments. Xiaojun Wan is the corresponding author.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *ICLR*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *EMNLP*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *NIPS, Deep Learning and Representation Learning Workshop*.

Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. 2017. Learning to generate product reviews from attributes. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 623–632, Valencia, Spain, April. Association for Computational Linguistics.

Wei Fang, Juei-Yang Hsu, Hung-yi Lee, and Lin-Shan Lee. 2016. Hierarchical attention model for improved machine comprehension of spoken content. *IEEE Workshop on Spoken Language Technology (SLT)*.

Pierre-Etienne Genest and Guy Lapalme. 2012. Fully abstractive approach to guided summarization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 354–358. Association for Computational Linguistics.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Alex Graves. 2012a. Sequence transduction with recurrent neural networks. *International Conference of Machine Learning (ICML) Workshop on Representation Learning*.

Alex Graves. 2012b. Supervised sequence labelling. In *Supervised Sequence Labelling with Recurrent Neural Networks*, pages 5–13. Springer.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137.

Soomin Kim, JongHwan Oh, and Joonhwan Lee. 2016. Automated news generation for tv program ratings. In *Proceedings of the ACM International Conference on Interactive Experiences for TV and Online Video*, pages 141–145. ACM.

Ioannis Konstas. 2014. Joint models for concept-to-text generation.

Jiwei Li, Minh-thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of ACL*. Citeseer.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of NAACL-HLT*, pages 110–119.

Zachary C Lipton, Sharad Vikram, and Julian McAuley. 2015. Generative concatenative nets jointly learn to write and classify reviews. *arXiv preprint arXiv:1511.03683*.

Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. 2014. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Jean Pouget-Abadie, Dzmitry Bahdanau, Bart van Merriënboer, Kyunghyun Cho, and Yoshua Bengio. 2014. Overcoming the curse of sentence length for neural machine translation using automatic segmentation. *Syntax, Semantics and Structure in Statistical Translation*, page 78.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. *ACL*.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 553–562. ACM.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. Lstm neural networks for language modeling. In *Interspeech*, pages 194–197.

Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 1017–1024.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Richard S Sutton and Andrew G Barto. 1998. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge.

Jian Tang, Yifan Yang, Sam Carton, Ming Zhang, and Qiaozhu Mei. 2016. Context-aware natural language generation with recurrent neural networks. *arXiv preprint arXiv:1611.09900*.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*.

Qixin Wang, Tianyi Luo, Dong Wang, and Chao Xing. 2016. Chinese song iambics generation with neural attention-based model. *arXiv preprint arXiv:1604.06274*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, volume 14, pages 77–81.

Rui Yan, Liang Kong, Congrui Huang, Xiaojun Wan, Xiaoming Li, and Yan Zhang. 2011. Timeline generation through evolutionary trans-temporal summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 433–443. Association for Computational Linguistics.

Xiaoyuan Yi, Ruoyu Li, and Maosong Sun. 2016. Generating chinese classical poems with rnn encoder-decoder. *arXiv preprint arXiv:1604.01537*.

Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2016. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4584–4593.

Jianmin Zhang, Jin-ge Yao, and Xiaojun Wan. 2016. Toward constructing sports news from live text commentary. In *Proceedings of ACL*.