Distinctive Slogan Generation with Reconstruction

Shotaro Misawa   Yasuhide Miura   Tomoki Taniguchi   Tomoko Ohkuma

Fuji Xerox Co., Ltd.

{misawa.shotaro,yasuhide.miura,
taniguchi.tomoki,ohkuma.tomoko}@fujixerox.co.jp

Abstract

E-commerce sites include advertising slogans along with information regarding items. Slogans can attract viewers' attention to increase sales or visits by emphasizing advantages of items. The aim of this study is to generate a slogan from a description of an item. To generate a slogan, we apply an encoder–decoder model which has shown effectiveness in many kinds of natural language generation tasks, such as abstractive summarization. However, slogan generation task has three characteristics that distinguish it from other natural language generation tasks: distinctiveness, topic emphasis, and style difference. To handle these three characteristics, we propose a compressed representation–based reconstruction model with refer–attention and conversion layers. The results of experiments with automatic and human evaluations indicate that our method achieves higher performance than conventional methods.

1   Introduction

Advertisements, e-commerce sites, and flyers include advertising slogans along with item information, such as descriptions, prices, features, and pictures. Although the purpose of item information is to tell accurate and detailed information, the aim of slogans is to attract viewers' attention to increase sales or visits. Therefore, a slogan often emphasizes a part of advantages that an item holds, and it often contains peculiar words or expression to draw viewers' attention.

Creating slogans is cumbersome and costly. We propose a method to generate a slogan from a description of a target item to support the writers of slogans. Figure 1 shows an example of a description and a slogan for a job matching. A slogan is similar to an abstractive summary to some extent; a target sentence is shorter than a description and includes words not presented in the description. However, to attract viewers, slogans should have three unique characteristics. **Distinctiveness**: A slogan should be distinctive towards the target item. When a slogan is generic and suitable also for any other items, it does not increase the viewer's motivation to select the target item. Therefore, a slogan should describe the detail of a target-item. For example, "Don't go with the flow, but make a new flow by developing business social media" is better than "Why don't you apply to our company? We need you right now!" because the former provides more information regarding the job. **Topic Emphasis**: A slogan should examine a specific topic in the description because it is difficult to contain all aspects within a short tip. In fact, the example slogan in Figure 1 does not include phrases such as "business social media," which is one of the main topics in the description. **Style Difference**: A slogan should be written in an impressive writing style to attract viewers' interests. Therefore, a slogan differs from a description in terms of style and vocabulary. For instance, the first sentence of the example slogan is not just an explanation but an interrogative sentence.

To generate a slogan, we apply an encoder–decoder model (Bahdanau et al., 2014) since encoder–decoder models have shown effectiveness in abstractive summarization, and slogan generation shares some characteristics with abstractive summarization. However, slogan generation has three characteristics that differentiate it from abstractive summarization: distinctiveness, topic emphasis, and style difference. In particular, summarization models lose distinctiveness because their outputs tend to be generic (Gimpel et al., 2013).

【Slogan】 How many times will you turn? We want marketers who will turn a PDCA-cycle at very high rate.

【Description】
We want to recruit marketing staffs for Wantedly. There are various products associated with the largest business social media, "*Wantedly*". Till date, we have registered with approximately 20,000 companies. We have developed a service that leads to "excitement at work" with the vision "We will be the infrastructure of all businesspeople in 2020!". To achieve that vision, we must operate a larger business at a higher speed than it currently is. We have many tasks for developing services. Will you join us to enhance the workplace for the future? Through work, you will turn a PDCA-cycle in a short period and maximize its effects! You cannot proceed if you don't try anything. Try "Code Wins Arguments," and scale new heights with *Wantedly*!

Business contents: KPI design and management. / Making hypotheses and plans based on analysis data. / Planning and implementation of promotion. / Advertisement operation. / Writing of recruitment. / Nurturing and lead generation by MA tools. / Improving services based on hearing to customers.

Graduates, people with programming experienced, and marketers who like new things are welcome. I would be glad to speak with you. Why not try new challenges with your colleagues and make a serious attempt at accomplishing something?

Figure 1: Example of a description and a slogan for a job-matching. Japanese–English translation.

To enhance distinctiveness, we define a reconstruction loss by introducing a reconstruction model that estimates the corresponding description from a slogan. The loss encourages the generation model to generate a distinctive slogan to help the estimation of the description. Here, slogan generation is a novel task in which information of a slogan is not equivalent to that of the description because of topic emphasis and style difference. Therefore, our reconstruction model generates a compressed representation of a description instead of the entire description. To ensure topic emphasis, we include a refer-attention layer that focuses on the partial description emphasized by a generated slogan into our model. To address style difference, we introduce a conversion layer that absorbs the style and vocabulary gaps between descriptions and slogans. We refer to this reconstruction model as a compressed representation–based reconstruction model (CRR).

The main contributions of our study can be summarized as follows.

- Propose a method that includes a compressed representation–based reconstruction model to generate distinctive slogans.
- Introduce the refer–attention and conversion layers for CRR to address topic emphasis and style difference, respectively.
- Evaluate the method using automatic and human evaluations.
- Compare the performance of non-neural and neural methods.

2 Related Work

Slogan Generation: Slogan generation has been addressed through non–neural approaches. One approach is to extract and revise a sentence from a corpus (Indrakanti et al., 2018; Iwama and Kano, 2018). Another approach is to construct skeletons of slogans and fill arbitrary words into the skeletons (Özbal et al., 2013; Alnajjar and Toivonen, 2020). In one study, suggestions for rephrasing written slogans were provided to assist slogan writers (Clark et al., 2018). Some approaches use a description of a target item (Žnidaršič et al., 2015; Chandler and Neupane, 2018), similarly to our method. These approaches consist from three steps: selecting slogans from a corpus as candidates, rephrasing candidate slogans, and sorting these slogans. However, the generated slogans are not creative because these methods essentially rephrase existing slogans. In contrast, our method is a neural–based generation model that generates slogans that are different from those in a corpus. The comparison of creativity between our method and a non-neural method is described in Section 5.3, in addition to the generation performance.

Abstractive Summarization: Summarization has been tackled by extractive and rule-based approaches (Vanderwende et al., 2007; Nenkova and McKeown, 2011; Nallapati et al., 2016a). However, neural based abstractive approaches have become very common and achieved high performance because of
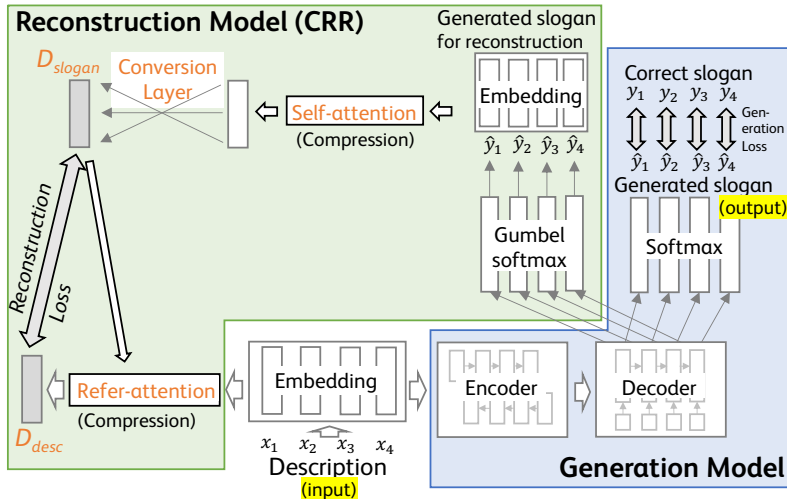
Figure 2: Outline of our method.

emerging large-scale datasets (Hermann et al., 2015) and powerful methods (Rush et al., 2015; Nallapati et al., 2016b; See et al., 2017) which are mainly based on an encoder–decoder model (Bahdanau et al., 2014). Because abstractive summarization is similar to slogan generation, neural abstractive methods will likely to improve the performance of slogan generation. On the other hand, these methods do not guarantee distinctiveness, topic emphasis, and style difference of generated texts which are required in slogans to improve sales or visits.

Reconstruction Model: Methods that include a reconstruction process have been proposed for text generation tasks. These methods combine a reconstruction model that generates a source sentence from information regarding a generated sentence, such as decoder's hidden states. Reconstruction models are used to improve the performance for low-resource data (Cheng et al., 2016) or application to monolingual data (Edunov et al., 2018) or multi-modal data (Delbrouck and Dupont, 2019). Some studies have introduced gumbel softmax (Niu et al., 2019) or reinforcement learning (Wang and Lee, 2018) to sample a generated sentence similar to our method. In addition, there is an approach to reconstruct the decoder's hidden states in video captioning (Wang et al., 2018). These methods reconstruct entire input sequences. However, this approach is inappropriate for slogan generation because of topic emphasis and style differences, thereby making a model difficult to reconstruct the entire description from a generated slogan. Our reconstruction model only generates a compressed representation, not a sentence or sequence, and the model includes mechanisms to ensure certain slogan characteristics.

3 Distinctive Slogan Generation with Reconstruction

A slogan should reflect a distinctive and important part of an item description. The more distinctive a slogan is, the easier the reconstruction of the corresponding description from the slogan is. This is because such slogans facilitate the estimation of the description. For example, the estimation of the corresponding description from the slogan "We need new staffs!" is difficult because it can fit to any other job. To enhance the distinctiveness of generated slogans, we define and minimize a reconstruction loss as an indicator of distinctiveness by introducing a compressed representation–based reconstruction model. To encourage topic emphasis and style difference, refer–attention and conversion layers are used, respectively. Figure 2 shows the outline of our method.

3.1 Generation Model

As a base generation model, any type of text generation models with an encoder–decoder model (Sutskever et al., 2014) can be applied. In our experiments, we employed two major approaches; an attentional encoder–decoder model (Bahdanau et al., 2014) and that with a copying mechanism (See et al., 2017). As training data, the base model requires pairs of a description and a correct slogan, and it is trained to generate the correct slogan from an input description.

89

Attentional Encoder-Decoder Model: Firstly, it divides a description into words. Secondly, it gets the embeddings of the description words. Here, we randomly initialize the embeddings and update these during training, however the embeddings can be pretrained. Thirdly, the embeddings are fed one-by-one into an encoder to calculate the context vector of the description. The encoder consists of a Recurrent Neural Network based model; specifically, a Gated Recurrent Unit (GRU) (Cho et al., 2014) is used.

Then, a decoder, which also consists of a Recurrent Neural Network based model, generates words from the context vector. It generates a word one-by-one with considering the previously generated word and the hidden representation of the decoder. The output probability of t-th word in a slogan, $P(\hat{w}_t)$, is calculable by equation 1.

$$P(\hat{w}_t) = \text{softmax}(W_G[h_t^G, C_t] + b_G) \tag{1}$$

$$h_t^G = \text{gru}(E(\hat{w}_{t-1}), h_{t-1}^G) \tag{2}$$

$$h_0^G = \tanh(W_0 ctx_I + b_0) \tag{3}$$

$$C_t = \sum_{i=1}^{I} a_i^t \cdot ctx_i \tag{4}$$

$$a_i^t = \text{softmax}(v \cdot \tanh(W_{att1} ctx_i + W_{att2} h_t^G + b_{att})) \tag{5}$$

where W_*, b_*, and v are trainable parameters, $[\cdot, \cdot]$ signifies the concatenation operation, $\hat{w}_t$ represents the t-th word of the predicted slogan, ctx_i denotes the context vector of the i-th word of the description, I is the length of the description, and $E(\cdot)$ signifies the embedding layer. In addition, softmax, tanh, and gru is a softmax, tangent hyperbolic, GRU function, respectively. The GRU function calculates the hidden states h_*^G from the previous hidden state of GRU and the previous word. During training, it uses the correct slogan word for $\hat{w}_{t-1}$ in equation 2. During inference, it uses beam search to find the best sentence.

Copying Mechanism: The decoder of an attentional encoder-decoder model is able to only generate words that exist in a predefined word vocabulary. To generate out-of-vocabulary words, this mechanism copies a word in an input description. Specifically, the probability of copying the words in a description follows the attention distribution in equation 5. That is, the word output probability is calculated from not only equation 1 but also from the attention distribution in equation 5. In addition, to balance these probabilities, it applies a parameter p_{gen}. Equations 6 and 7 show the calculation of the output probability of the t-th word in a slogan, $P_{copy}(\hat{w}_t)$.

$$P_{copy}(\hat{w}_t = w) = p_{gen}^t P(\hat{w}_t = w) + (1 - p_{gen}^t) \sum_{i:x_i=w} a_i^t \tag{6}$$

$$p_{gen}^t = \sigma(W_{pt1} ctx_* + W_{pt2} h_t + W_{pt3} E(\hat{w}_{t-1}) + b_{pt}) \tag{7}$$

where σ represents the sigmoid function, and x_i denotes the i-th word of a description.

3.2 Reconstruction Model with Refer-Attention and Conversion Layer

The reconstruction model is used to estimate the description from a generated slogan. To handle information inequivalence between descriptions and slogans, the model compresses a description and a generated slogan into representations, respectively. The reconstruction loss is calculated as the difference between the representation of a generated slogan (D_{slogan}) and that of a description (D_{desc}).

Generated Slogan Representation (D_{slogan}): The representation of a generated slogan can be obtained with word embeddings, but the standard generation process, such as beam search, is not differentiable. Therefore, we apply a gumbel–softmax layer (Jang et al., 2017) for which the input is the output from the generation model decoder (e.g. the output from equation 1). Equation 8 represents the process of sampling a slogan.

$$ss_t = gumbel(d_t) \tag{8}$$

where ss_t represents the t-th word of a sampled slogan, d_t signifies the decoder output of the t-th word, and $gumbel$ denotes the gumbel–softmax layer.

Then, the generated slogan representation is calculated from a weighted average of the word embeddings of the generated slogan which is sampled in equation 8. Here, slogans often contain words aiming at attracting notice and having no specific meaning, and these words are unnecessary for a reconstruction. Therefore, the words in slogans are weighted through the self–attention layer α_t^{self} inspired by the previous work (Vaswani et al., 2017) to ignore unnecessary words. The weighted representation Rep_{slogan} is calculated by equation 9.

$$Rep_{slogan} = \sum_{t=1}^{T} \alpha_t^{self} E(ss_t) \tag{9}$$

$$\alpha_t^{self} = \frac{\exp(att_{base}^t)}{\sum_{j=1}^{T} \exp(att_{base}^j)} \tag{10}$$

$$att_{base}^t = W_{a1}(\tanh(W_{a2}\{E(ss_t) + b_{a2}\}) + b_{a1} \tag{11}$$

where T signifies the length of a generated slogan.

There is style difference between slogans and descriptions, such as word usage or writing style. To learn and adjust style difference, a conversion layer is applied. Equation 12 shows the representation of a generated slogan with this conversion layer.

$$D_{slogan} = W_c(Rep_{slogan}) + b_c \tag{12}$$

Description Representation (D_{desc}): The representation of a description is defined by a weighted average of the word embeddings of the words in a description. Here, a slogan pays attention to a specific part of the description because of topic emphasis. Therefore, we introduce the refer–attention layer α_i^{refer} to examine specific description words that a generated slogan emphasizes. The description representation D_{desc} is calculable by equation 13.

$$D_{desc} = \sum_{i=1}^{I} \alpha_i^{refer} \times E(x_i) \tag{13}$$

Since the refer–attention layer should be able to find parts which a generated slogan focused on, α_i^{refer} should be calculated by considering both the slogan representation D_{source} and a word in the description. Equation 14 presents the calculation of the refer attention for the i-th word of a description.

$$\alpha_i^{refer} = \frac{\exp(v_r u_i)}{\sum_{j=1}^{I} \exp(v_r u_j)} \tag{14}$$

$$u_i = \tanh(W_{r1}D_{slogan} + W_{r2}E(x_i) + b_r) \tag{15}$$

where v_r signifies the trainable parameter.

3.3 Objective Function

To optimize the generation and reconstruction models, we define generation and reconstruction losses, respectively. When training the both models, we optimize the total loss that is combined loss of the generation and reconstruction losses. When combining, it multiplies λ by the reconstruction loss to balances these losses. Equation 16 shows the total loss L.

$$L = L_1 + \lambda \times L_2 \tag{16}$$

Generation Loss: The generation loss is defined as the cross-entropy loss between generated and correct slogans. Equation 17 shows the generation loss L_1.

$$L_1 = -\sum_{k=1}^{K} \log \hat{c}_k \tag{17}$$

	Train	Dev.	Test
# pairs of a description and a slogan	58,461	6,290	7,573
avg. slogan length	15.59	15.38	15.57
avg. description length	347.14	314.75	362.67

Table 1: Data specifications

Dataset	uni-gram	tri-gram
Abstractive Summary	68.4%	17.3%
Slogan	63.0%	8.7%

Table 2: The average rate of a target sentence uni- and tri-grams that exist in the corresponding source sentence

where K signifies the length of a correct slogan and $\hat{c}_k$ represents the generation probability of the k-th word of the correct slogan.

Reconstruction Loss: The reconstruction loss should become lower when D_{desc} and D_{slogan} are similar. In our experiment, the reconstruction loss is defined by using the cosine similarity between the representations of generated and correct slogans. Equation 18 shows the reconstruction loss L_2.

$$L_2 = 1 - cos(D_{desc}, D_{slogan}) \tag{18}$$

where $cos(D_{desc}, D_{slogan})$ represents the cosine similarity between D_{desc} and D_{slogan}.

4 Experimental Settings

4.1 Dataset

We used a dataset collected from *Wantedly*[1], a Japanese job matching website. The data includes many pairs of slogans and descriptions, which is a requirement of this study. The descriptions include job details and requirements.

Each job has the corresponding company, and each company has 10.2 jobs on average. Slogans of the same company tend to become similar to each other. Therefore, data were split into three sets with the ratio of 8 : 1 : 1, provided that the data of the same company belong to the same set. Table 1 presents the data specifications.

We compared this slogan dataset with a Japanese newspaper corpus of abstractive summary [2]. Table 2 represents the rate of a target sentence (e.g. slogan) uni-gram and tri-gram that appear in the source sentence (e.g. description). By comparing abstractive summaries and slogans, slogans contain more words that do not appear in the source sentence. Furthermore, tri-grams of slogans are more unseen in the source sentence than that of summaries. These results indicate that the writing style of slogans is different from that of descriptions, thus making slogan generation more difficult than abstractive summarization.

4.2 Comparison Methods

Non-neural Method: As a non–neural method, we developed Case–based slogan production (Žnidaršič et al., 2015) (Case–Base). It retrieves a relevant case from training data and rewrites some words by considering part of speech tags.

Neural Baseline Methods: As neural baselines, we prepared an attentional encoder–decoder (EncDec) model and that with a copying mechanism (Copy).

Conventional Method of Reconstruction: To compare our method with the method of reconstructing an entire sentence, we developed a method in which the reconstruction model generates a full description from a slogan by introducing an EncDec model (Niu et al., 2019) (EncDec+RecText).

Proposed Methods: We prepared our methods with the generation model of EncDec (EncDec+CRR) and Copy (Copy+CRR).

Other Configurations of Proposed Method: To investigate the effect of the attention layer in CRR, we replaced the refer–attention layer with an average pooling layer (EncDec+CRR-woAtt). To examine the effect of the conversion layer, we developed a CRR without the conversion layer (EncDec+CRR-woConv).

---

[1]https://www.wantedly.com
[2]https://mainichi.jp/contents/edu/03.html

	ROUGE-L
Case-Base (Žnidaršič et al., 2015)	13.41
EncDec (Bahdanau et al., 2014)	16.77
EncDec+RecText (Niu et al., 2019)	17.14
EncDec+CRR-woAtt	17.61
EncDec+CRR-woConv	17.89
EncDec+CRR	**18.10**
Copy (See et al., 2017)	18.50
Copy+CRR	**19.38**

Table 3: Automatic evaluation results. Bold indicates the best score for each baseline.

	Distinc.	Adeq.	Flu.
Copy	47.5	49.8	50.0
Copy+CRR	**52.5**	**50.2**	50.0

Table 4: Human evaluation results. Bold indicates the best score for each measure.

4.3 Configurations

Data Preprocess: We used NEologd (Sato et al., 2017) for word segmentation. We removed non-frequent words which was over 50,000 words in frequent order.

Hyperparameters: We run five experiments for each parameter and selected λ and p which achieved the best ROUGE-L performance for the development data. λ was selected from $\{0.5, 1.0, 2.0, 3.0, 4.0, 5.0\}$. The value of λ was set to 4.0 with the performance of EncDec+CRR. The value of p was set to 0.6 from $\{0.1, 0.2, 0.3, \cdots, 0.8, 0.9, 1.0\}$. The other parameters were decided by the performance of EncDec. The dimensions of embeddings and GRU were 200. In addition, Adam optimizer with the default parameter of pytorch was used for optimization, and the beam size of inference was 5.

4.4 Evaluation Metrics

Automatic Evaluation: We run 10 experiments and used the average of ROUGE-L (Lin, 2004) which is a popular metric of language generation tasks. ROUGE-L is the longest common sequence-based statics. We calculated ROUGE score using the public python script of rouge [3] (version 0.3.0).

Human Evaluation: We conducted a pairwise comparison of Copy and Copy+CRR via crowdsourcing. Both methods generated slogans for 250 randomly sampled descriptions from the test data. Each worker judges the generated slogans in terms of distinctiveness (Distinc.), adequacy (Adeq.), and fluency (Flu.) by looking at a description of a target job and slogans generated by Copy and Copy+CRR. We took the majority vote of evaluations of 10 workers. Distinctiveness is a measure of how specialized a slogan is to the corresponding description. Adequacy is that of how faithful a slogan is to the description. Fluency refers to how natural a slogan is.

5 Results and Discussion

5.1 Automatic Evaluation Result

Table 3 presents the automatic evaluation results. Note that slogans are creative, therefore the ROUGE-L performance itself is relatively lower than other tasks.

Non-neural method vs Neural methods: The results indicate that all of the neural methods outperform Case-Base which is the non–neural conventional method. Moreover, the performance gap between non-neural and neural methods was quite large.

Baseline method vs Proposed method: When comparing EncDec with EncDec+CRR and Copy with Copy+CRR, the methods with CRR outperformed its baseline method. This result indicates that our reconstruction model improves the performance of slogan generation.

RecText vs CRR: EncDec+RecText also achieved higher performance than EncDec; however, its performance was worse than that of EncDec+CRR. This result suggests that reconstructing the entire description is inappropriate for slogan generation. Here, we observed that the descriptions reconstructed

---

[3]https://github.com/pltrdy/rouge

Methods	Slogan
Correct	How many times will you turn? We want a marketer who will turn a PDCA-cycle at very high rate.
Copy	We want a person who will be in charge of marketing of the future!
Copy+CRR	We want a marketer who will make a new service at the speed of light!

Table 5: Example outputs. The description is presented in Figure 1.

Methods	Creativity
Case-Base	59.67
Copy+CRR	69.81

Table 6: Creativity comparison between non-neural and neural methods

Methods	Distinctiveness
Copy	75.34
Copy+CRR	71.27

Table 7: Distinctiveness comparison between baseline and our methods

by RecText were quite poor. We have concluded that the reconstruction loss in RecText tends to focus on generating a fluent description than enhancing the distinctiveness of a slogan.

Effect of Each Component of CRR: Both the refer–attention and conversion layers also had positive effects when comparing CRR, CRR-woAtt, and CRR-woConv. In particular, the effect of the refer–attention layer was greater than that of the conversion layer. One reason for this may be that style difference is absorbed by not only the conversion layer but also the self–attention layer.

5.2 Human Evaluation Result

Table 4 presents the results of the human evaluation. Our method performed better in terms of distinctiveness; this illustrates that CRR is effective in reflecting distinctive information of a description. Adequacy of Copy+CRR was slight better than that of Copy; it represents that our method can capture correct information of the target job even though generic slogans (i.e. less distinctiveness) have less risk of containing wrong information. In terms of Fluency, the two methods were equal. It indicates that CRR does not adversely affect the language modeling of the generation model.

5.3 Discussion

Example Outputs: Table 5 presents the correct slogan and outputs from Copy and Copy+CRR of which descriptions are presented in Figure 1. The output from Copy includes "marketing" as a word related to the description. The output from Copy+CRR includes "speed of light" in addition to "marketer"; it indicates that our method can describe more distinctive information of a description. Moreover, "speed of light" is a metaphor and does not naturally appear in a job description. Such expression in slogans might draw attention of readers. In terms of drawing attention, the generated slogan of Copy+CRR seems better than the expression "at very high rate" in the correct slogan. This might be because the generation model prefers extreme words or phrases to make it easier to distinguish the target job from another job. However, it is still difficult to generate the first sentence of the correct slogan.

Creativity of Neural Method: To confirm that neural methods can generate more creative slogans than non-neural methods, we measured creativity of the slogans generated by Case-Base and Copy+CRR. Here, we define creativity as the similarity of the generated slogans and the training data. We measured ROUGE-L between a generated slogan and the slogan that is most similar to the generated slogan in the training data. Table 6 represents the average score of Case-Base and Copy+CRR. The result indicates that our neural method is more creative than the conventional non–neural method.

Distinctiveness of Outputs: To verify that our method can generate more distinctive slogans than a conventional method, we calculated distinctiveness of the generated slogans. As an alternative measure of distinctiveness, we assessed the similarity of the slogans generated for the target job and another job. This is because the generated slogans become dissimilar when each slogan is distinctive towards the target job, given that each job is different each other. We calculated the averaged ROUGE-L between each generated slogan and the most similar generated slogan in the test data. Table 7 illustrates the score
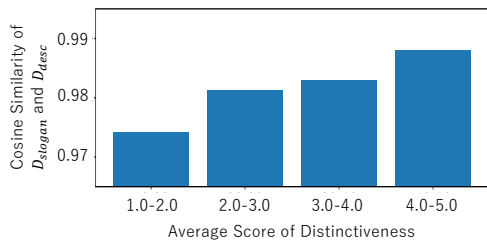
Figure 3: Relationship between distinctiveness and cosine similarity of representations
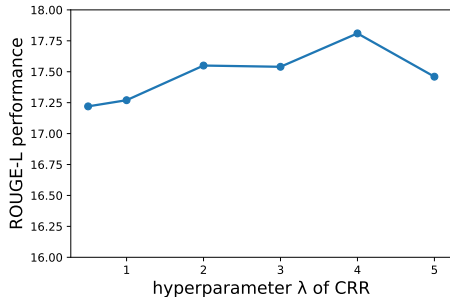

Figure 4: Performance curve of different λ of EncDec+CRR

of Copy and Copy+CRR. The result suggests that the outputs from Copy+CRR are more dissimilar from each other than the outputs from Copy. Therefore, CRR induces distinctiveness in the generated slogans.

Effect of Reconstruction Model: By updating the reconstruction model, the reconstruction loss could be reduced. To check whether the generation model changes output tips to reduce the reconstruction loss, we evaluated the reconstruction loss when the loss is backpropagate to only the reconstruction model (OnlyR) and to the both reconstruction and generation models (Both). That is, OnlyR was trained to estimate the description from the generated slogan which is an intact output from the generation model. The reconstruction loss for Both was 0.006, however that for OnlyR was 0.112. This result indicates that the reconstruction loss makes the generation model generate slogans which can be a clue to the estimation of the description.

To authorize the assumption that the reconstruction is easy when generated slogans are distinctive, we compared distinctiveness of generated slogans and the ease of the reconstruction of the description from a slogan. As a measure of distinctiveness, workers in human evaluation evaluated 250 slogans generated by Copy+CRR in terms of distinctiveness on a scale from 1 to 5. As a measure of the ease of the reconstruction, we used $cos(D_{desc}, D_{slogan})$ in the reconstruction loss when generating the slogan. Figure 3 presents the relationship between the averaged distinctiveness scores and the cosine similarities. This result suggests that the generated slogan becomes distinctive as it can estimate the description of the target job accurately. That is, it supports our assumption that the difficulty of reconstruction is related to the distinctiveness of generated slogans.

Effect of Hyperparameter λ: To analyze the effect of λ, we examined the relationship between the performance and λ. Figure 4 shows the performance curve of different λ. The figure shows that the best λ is 4.0. However, the variation in the performance for λ is relatively small. It indicates that the parameter λ is not sensitive, and our method does not hurt the performance if choosing wrong λ.

Note that the best λ is over 1.0, therefore the method seems to emphasize L_2 in equation 16. However, the value of the reconstruction loss is very small, and the method mainly focuses on the generation loss while referring to the reconstruction loss.

6 Conclusion

The aim of this study is to generate a slogan from the description of a target item. We firstly apply an encoder–decoder model to the slogan generation task. Then, we focus on the three characteristics of slogan generation: distinctiveness, topic emphasis, style difference. To enhance the distinctiveness of generated slogans, we define and minimize a reconstruction loss as an indicator of distinctiveness by introducing a compressed representation–based reconstruction model. To encourage topic emphasis and style difference, refer–attention and conversion layers are used, respectively. Automatic and human evaluation demonstrated that our method is superior to the conventional methods. In addition, we observed that neural methods are superior to a non-neural method in terms of ROUGE-L performance and creativity. In future work, we are planning to make a generation model which can directly enhance sales or visits by optimizing more straightforward e-commerce signals. Moreover, we will like to conduct an experiment with another dataset to ensure the generalizability of our method.

References

Khalid Alnajjar and Hannu Toivonen. 2020. Computational generation of slogans. *Natural Language Engineering*, pages 1–33.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bryant Chandler and Aadesh Neupane. 2018. Slogatron: Advanced wealthiness generator. *arXiv preprint arXiv:1809.09563*.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *Eight Workshop on Syntax, Semantics and Structure in Statistical Translation*.

Elizabeth Clark, Anne S. Ross, Chenhao Tan, Yangfeng Ji, and Noah A. Smith. 2018. Creative writing with a machine in the loop: Case studies on slogans and stories. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces*, pages 329–340.

Jean-Benoit Delbrouck and Stephane Dupont. 2019. Adversarial reconstruction for multi-modal machine translation. *arXiv preprint arXiv:1910.02766*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500.

Kevin Gimpel, Dhruv Batra, Chris Dyer, and Gregory Shakhanarovich. 2013. A systematic exploration of diversity in machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1100–1111.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, page 1693–1701.

Saratchandra Indrakanti, Gyant Singh, and Justin House. 2018. Blurb mining: Discovering interesting excerpts from e-commerce product reviews. In *Companion Proceedings of the The Web Conference 2018*, pages 1669–1675.

Kango Iwama and Yoshinobu Kano. 2018. Japanese advertising slogan generator using case frame and word vector. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 197–198.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *Proceedings of the International Conference on Learning Representations*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of The Workshop on Text Summarization Branches Out at the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 74–81.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2016a. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *arXiv preprint arXiv:1611.04230*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016b. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Ani Nenkova and Kathleen McKeown. 2011. *Automatic summarization*. Now Publishers Inc.

Xing Niu, Weijia Xu, and Marine Carpuat. 2019. Bi-directional differentiable input reconstruction for low-resource neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 442–448.

Gözde Özbal, Daniele Pighin, and Carlo Strapparava. 2013. Brainsup: Brainstorming support for creative sentence generation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1446–1455.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Toshinori Sato, Taiichi Hashimoto, and Manabu Okumura. 2017. Implementation of a word segmentation dictionary called mecab-ipadic-neologd and study on how to use it effectively for information retrieval (in japanese). In *Proceedings of the Twenty-three Annual Meeting of the Association for Natural Language Processing*, pages NLP2017–B6–1. The Association for Natural Language Processing.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083.

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

Martin Žnidaršič, Polona Tomašič, and Gregor Papa. 2015. Case-based slogan production. In *Proceedings of The Workshop on Experience and Creativity at the 23rd International Conference on Case-Based Reasoning*, pages 123–130.

Lucy Vanderwende, Hisami Suzuki, Chris Brockett, and Ani Nenkova. 2007. Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N.Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st Conference on Neural Information Processing Systems*.

Yaushian Wang and Hung-Yi Lee. 2018. Learning to encode text as human-readable summaries using generative adversarial networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4187–4195.

Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. 2018. Reconstruction network for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7622–7631.