Review Headline Generation with User Embedding

Tianshang Liu[1,2], Haoran Li[1,2], Junnan Zhu[1,2], Jiajun Zhang[1,2], and Chengqing Zong[1,2,3]

[1] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai 200031, China
{tianshang.liu,haoran.li,junnan.zhu,jjzhang,cqzong}@nlpr.ia.ac.cn

Abstract. In this paper, we conduct a review headline generation task that produces a short headline from a review post by a user. We argue that this task is more challenging than document summarization, because the headlines generated by users vary from person to person. It not only needs to effectively capture the preferences of the users who post the reviews, but also requires to mine the emphasis of the users regarding the review when they write the headlines. To this end, we propose to incorporate the user information as the prior knowledge into the encoder and decoder for general sequence-to-sequence model. Specifically, we introduce user embedding for each user, and then we use these embeddings to initialize the encoder and decoder, or as biases for decoder initialization. We construct a review headline generation dataset, and the experiments on this dataset demonstrate that our models significantly outperform baseline models which do not consider user information.

Keywords: Review Headline Generation · User Embedding · Sequence-to-sequence Neural Network.

1 Introduction

Review headline generation is a task that aims to generate a headline for a user review. A user review of an item generally provides user experience for this item, including user satisfaction, and user preferences. Therefore, review headline generation is very essential in helping potential users of an item to quickly get the gist of the original review. In this work, we focus on generating a headline for a single-review.

Review headline generation is a special case of document summarization [22, 25, 30, 15]. Different from document summarization, headlines generated by users are highly subjective. In other words, a review headline is not only correlated to the textual information of the review, but is also influenced by user preference. However, we find that previous document summarization systems almost ignore user information. Thus, for review headline generation task, we believe that it is necessary to design a novel model that can capture the preference of the user and to mine the emphasis of the users regarding the review when they write the headlines.

To this end, we propose a method generating review headline with user embedding, which regards the user information as the prior knowledge. Specifically, for each user, we introduce a user embedding, and then we use user embedding to initialize the encoder and decoder, or as a bias for decoder initialization. To verify our idea, we construct a dataset for hotel review, and the experiments on this dataset demonstrate that our models significantly outperform baseline models which do not consider user information.

Our main contributions are as follows:

- We propose a single-review headline generation framework that incorporates user information into the attention-based sequence-to-sequence (seq2seq) framework.
- Experimental results show that using user embedding as the prior knowledge can significantly boost the performance of our model, which verifies our hypothesis on the essential role of user information for review headline generation task.

2 Related Work

Our work is mainly related to abstraction-based text summarization and natural language processing tasks using user information.

2.1 Abstractive Document Summarization

Abstractive document summarization generates a summary of the source document by building the semantic representation of a document. The generated summary may contain some words or sentences that not presented in the original document. The core of the abstractive document summarization lies in exactly representing the semantics of the original document, and then accurately attending the critical parts of the source document, and finally generating the gist of the source document. Comparing to the extraction-based summarization methods [10, 28, 2] that select important sentences or phrases from the source document, abstraction-based summarization [23, 14, 8, 7, 27, 25, 12] involves rewriting summary as a human-written summary usually does.

In recent years, researchers employ seq2seq framework to tackle the abstraction-based text summarization problem. [22] proposed a neural network based model with local attention modeling, which is trained on the Gigaword corpus, but combined with an additional log-linear extraction-based summarization model with hand-crafted features. [6] introduced a conditional recurrent neural network that acts as a decoder to generate the summary of an input sentence and at each time-step the decoder also takes a conditional input which is the output of an encoder module. [18] proposed to control the vocabulary size to improve the training efficiency. [23] augmented the seq2seq model by introducing soft copy mechanism for reproduction of information while retaining the ability to generate novel words. [30] extended the seq2seq framework and proposed a selective gate network to control the information flow from the encoder to decoder. [15] equipped the seq2seq oriented encoder-decoder model by adding a deep recurrent generative decoder which is used to learn the latent structure information implied in the target summaries.

Compared to those researches mentioned above, we argue that review headline generation is more challenging because users have their own preferences when writing the headlines for the reviews they post. Thus, we need to consider user information to generate a headline for the review.

2.2 User Information Used in Natural Language Processing Tasks

In other user-related natural language processing tasks, some researches have considered user information.

[21] and [27] used “neighbor” users’ reviews to extend the traditional recommendation system. [4] proposed an approach to construct vectors to represent profiles of users and items under a unified framework to maximize word appearance likelihood. Then, the vectors were used for a recommendation task in which they predicted scores on unobserved user-item pairs without given text. [13] presented a weakly supervised approach to extract user attributes from user-generated text on Twitter which can provide the social connection information. [29] proposed two different neural networks to learn both user embeddings and text embeddings for scholarly Microblog recommendation. [3] presented a novel framework which used only the user id and user social contexts for gender prediction. Their key idea is to represent users in the embedding connection space by considering users’ social contexts including family members, schoolmates, colleagues, and friends. [1] proposed to use user embeddings as lexical signals to recognize sarcasm. [26] proposed an embedding approach to learning user profiles, where users were embedded on a topical interest space, and then they directly utilized the user profiles for search personalization. [16] proposed mixture models which exploited user and item embedding in latent factor models for recommendations. [19] introduced an embedding model based on capsule network to model the 3-way (query, user, document) relationships for search personalization. In their model, each user was embedded as a vector in the same vector space as words.

In our task, headlines generated by users are different from person to person. However, we find that previous document summarization systems almost ignore user information. Thus, for review headline generation task, we believe that it is necessary to design a novel model that can take advantage of user information. To this end, we propose a method to generate review headline with user embedding, which regards the user information as the prior knowledge.

3 Background: Seq2Seq Model

In this section, we describe the basic seq2seq learning framework. Given a dataset of review-headline pairs, $\mathcal{D} = (x_i, y_i^*)_i^N$, the seq2seq model maximizes the conditional probability of a target sequence $y^* : p(y^*|x)$. Recurrent Neural Networks (RNNs) encoder [5, 9] reads and converts a variable length input sequence x into a context representation c as follows:

$$h_t = f_{enc}(x_t, h_{t-1}) \tag{1}$$

$$c_t = f_c(h_t, ..., h_t) \tag{2}$$

where h_t is the hidden state at time-step t. c_t is a context vector generated from the sequence of the hidden states. f_{enc} and f_c are nonlinear activation functions.

The decoder generates word y_t given the context vector c_t and the previously generated words $\{y_1,...,y_{t-1}\}$:

$$p(y_t|y_1, ..., y_{t-1}) = f_{dec}(y_{t-1}, s_t, c_t) \tag{3}$$

where s_t is the hidden state of the decoder and f_{dec} is a nonlinear activation function that computes the probability vector for output words at time-step t. The maximum likelihood (ML) framework tries to minimize negative log-likelihood loss of the parameters.

$$\mathcal{L}_{ML}(\mathcal{D}) = \sum_{(x,y^*)\in\mathcal{D}} -\log\, p(y^*|x) \tag{4}$$

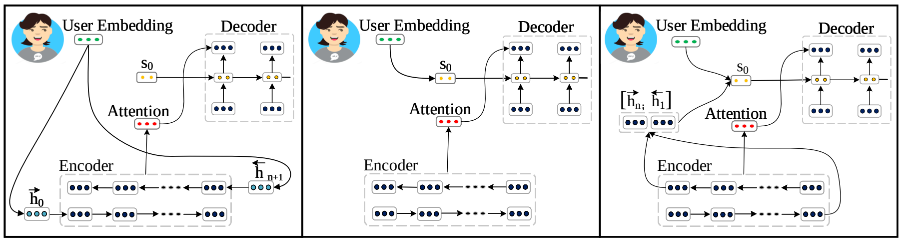
4 Our Model

4.1 Overview

We begin by introducing the review headline generation task. The input of the task is a review that is post by a user, and the output is a headline for this review. The sentence encoder is a bidirectional LSTM [9] (BiLSTM). Our summary decoder is a uni-directional LSTM with an attention mechanism and a softmax layer over the target vocabulary to generate words. Specifically, beyond general seq2seq architecture, we explore the information of user who posts the review to navigate our model to generate a headline that is unique to that user. To this end, we design multiple strategies to incorporate user into our model, namely, user-specific encoder initialization, user-specific decoder initialization, user-bias decoder initialization, as depicted in Figure 1. Furthermore, we combine above-mentioned strategies to initialize the encoder and decoder with the help of user information.

4.2 User-specific Encoder Initialization

In the general attention-based seq2seq model, the initial hidden states of the encoder, $\overrightarrow{h}_0$ and $\overleftarrow{h}_{n+1}$, are initialized by the zero vectors. In this work, we



(a) User-specific Encoder Initialization (b) User-specific Decoder Initialization (c) User-bias Decoder Initialization

Fig. 1. Framework of our model. We introduce user embedding for each user, and then we use these embeddings to initialize the encoder (a) , initialize the decoder (b), or as biases for decoder initialization (c).

propose a user-specific encoder initialization strategy which initializes the hidden states of the encoder by the user embedding. In details, we use two single-layer feed-forward neural networks to compute the initial hidden states of the forward and backward encoder, respectively, as follows:

$$\overrightarrow{h}_0 = \tanh(\mathbf{W}_p u + b_p) \tag{5}$$

$$\overleftarrow{h}_{n+1} = \tanh(\mathbf{W}_q u + b_q) \tag{6}$$

where $\mathbf{W}_p$, $\mathbf{W}_q$, b_p, and b_q are model parameters, and u denotes user embedding.

The idea behind this strategy is that different users have different preferences when they post the review, and we intend to explore the user preferences as the prior knowledge for the encoder.

4.3 User-specific Decoder Initialization

Generally, the initial hidden state of the decoder, s_0, is initialized by concatenating forward and backward vectors for the last hidden states of encoder as follows:

$$s_0 = \tanh(\mathbf{W}_h[\overrightarrow{h}_n; \overleftarrow{h}_1] + b_h) \tag{7}$$

where $\mathbf{W}_h$ and b_h are model parameters.

In this work, to incorporate user information into the decoder, we propose a user-specific decoder initialization strategy which initializes the hidden state of the decoder by the user embedding. Specifically, we use a single-layer feed-forward neural networks to calculate the initial hidden state of the decoder instead of the one in Equation 7, as follows:

$$s_0 = \tanh(\mathbf{W}_u u + b_u) \tag{8}$$

where $\mathbf{W}_u$ and b_u are model parameters.

The motivation for this strategy is that different users have different emphasis when they write the headlines for the review post by themselves, and we mine the emphasis regarding the whole review, and we argue that the user-specific emphasis can provide the prior knowledge for the decoder.

4.4 User-bias Decoder Initialization

In this strategy, instead of initializing decoder only by the user information as in Equation 8, we include the user embedding as an additional input to initialize the decoder hidden state as follows:

$$s_0 = \tanh(\mathbf{W}_v[\overrightarrow{h}_n; \overleftarrow{h}_1] + \mathbf{W}_w u + b_v) \tag{9}$$

where $\mathbf{W}_v$, $\mathbf{W}_w$, and b_v are model parameters.

In this way, the decoder initialization is related to both the source and user information. In other words, user-specific emphasis is used as a bias to the source representation.

4.5 Initializing the Encoder and Decoder with User Embedding

To jointly incorporate user information into the encoder and decoder, we combine (1) user-specific encoder initialization and user-specific decoder initialization; (2) user-specific encoder initialization and user-bias decoder initialization. We argue that these two strategies can simultaneously regard user information as prior knowledge for the encoder and decoder, which are more powerful than other strategies.

5 Experiments

5.1 Dataset

We construct the hotel review dataset by crawling data from TripAdvisor[4], which is a travel review website including review, headline, and the user who posts them. We finally get 315,396 user-review-summary triples. Statistics and division of the dataset are shown in Table 1.

Table 1. Dataset Statistics.

#users	#reviews	#headlines	#reviews/user	#words/review	#words/headline
6,682	315,396	315,396	47.2	158	7.7
Dataset			train	valid	test
#Review-Headline			305,396	5000	5000

[4] https://www.tripadvisor.com/

5.2 Comparative Methods

We compare the following methods.

- **Lead** baseline uses the first sentence in the review as the headline.
- **ABS** [22] uses an attentive CNN encoder and a neural network language model decoder to summarize the sentence.
- **SEASS** [30] is a state-of-the-art sentence summarization systems, which employs a selective encoding model to control the information flow from the encoder to the decoder.
- **Seq2seq** model is a standard attention-based seq2seq model without using user information.
- **Seq2seq+USE** model is a seq2seq model with user-specific encoder initialization.
- **Seq2seq+USD** model is a seq2seq model with user-specific decoder initialization.
- **Seq2seq+UBD** model is a seq2seq model with user-bias decoder initialization.
- **Seq2seq+USE+USD** model is a seq2seq model with user-specific encoder initialization and user-specific decoder initialization.
- **Seq2seq+USE+UBD** model is a seq2seq model with user-specific encoder initialization and user-bias decoder initialization.

5.3 Experimental Settings

We initialize model parameters by uniform randomly with range $[-0.1, 0.1]$. We set the size of word embedding and user embedding to 300 and LSTM hidden state size to 512. We use the full source and target vocabularies collected from the training data, which have 119,602 and 18,520 words, respectively. We use dropout [24] with probability of 0.2 and gradient clipping [20] with range $[-5, 5]$. We set the initial learning rate for Adam [11] to 5×10^{-4}, $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. we use mini-batch size 64 by grid search. At training time, we test ROUGE-2 F1 score on the development set for every 2,000 batches, and we halve the learning rate if model performance drops. Our models typically converge within 20 epochs using an early stopping strategy. At test time, we use beam search with beam size 10 to generate the summary.

5.4 Evaluation Metric

We employ ROUGE [17] as our evaluation metric. ROUGE measures the quality of summary by computing overlapping textual units including unigram, bigram, longest common subsequence (LCS), and skip bigram, which is a standard evaluation metric for summarization tasks. In this work, we report F1 score for ROUGE-1 (unigram), ROUGE-2 (bi-gram), ROUGE-L (LCS), and ROUGE-SU4 (bigram that allows for a skip distance up to 4 words) in our experimental results.

Table 2. Main experimental results (%). Our seq2seq models with user information perform significantly better than baseline models by the 95% confidence interval in the ROUGE script.

Model	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU4
Lead	13.19	3.20	11.50	2.39
ABS [22]	15.64	4.83	14.87	3.71
SEASS [30]	18.41	5.33	17.41	4.19
Seq2seq	17.93	5.32	16.88	4.24
Seq2seq + USE	19.95	6.40	18.85	5.19
Seq2seq + USD	20.83	6.68	19.59	5.59
Seq2seq + UBD	20.25	6.38	19.00	5.37
Seq2seq + USE + USD	**20.83**	**6.79**	**19.61**	**5.61**
Seq2seq + USE + UBD	20.12	6.45	18.90	5.35

5.5 Experimental Results

Table 2 shows the results of the comparative methods and our proposed methods. The models in the first portion of Table 2 are widely compared baseline methods. The state-of-the-art sentence summarization systems **SEASS** performs better than other baselines and our **seq2seq** model. Our proposed models considering user information achieve better performances than other models, which proves that user information plays a significant role for review headline generation task. **User-specific decoder initialization** strategy is more effective than **user-specific encoder initialization**, and we argue that a better decoder initialization is more important because decoder initialization is directly related to the generated words in the headlines. **User-bias decoder initialization** strategy performs worse than **user-specific decoder initialization**, which may attribute to that the source representation dilutes the influence of user information when initializing the decoder. Combining **user-specific encoder and decoder initialization** leads to a further improvement, which achieves +2.90% ROUGE-1, +1.47% ROUGE-2, +2.73% ROUGE-L, and +1.37% ROUGE-SU4 improvements over the basic **seq2seq** model.

5.6 Case Study

In this section, we show several review-headline pairs post by a specific user. For the review-headline pair in the test set, we show comparisons of the reference headline and the headlines generated by **seq2seq model**, **seq2seq model with user-specific encoder and decoder initialization**. From Figure 2, we can conclude that user have specific preferences when they post the reviews and write the headlines, and our model considering user information successfully captures these preferences while baseline model fails.

6 Conclusion

This paper addresses a review headline generation task, namely, how to produce a headline for a review post by a user. We prove that user information as the prior knowledge is vital for this task. Our proposed models explore the effectiveness of the user embedding to initialize the encoder and decoder, and the experiments on our constructed review headline generation corpus show our proposed framework significantly outperforms the baseline models which do not consider user information. We conclude that decoder initialization with only user embedding is a more valid strategy than others. When we simultaneously initialize the encoder and decoder using user embedding, the performance is further improved.

Review-headline pairs (in the training set) post by User-A:
Review 1: this is a solid 3 rating (out of 5) . *right across the street from the stadium and university . many restaurants only 1 minute walk .* very clean hotel (...) small selection but ok for the price of the room .
Headline 1: *good location to the stadium .* very low price . very nice staff
Review 2: they have an excellent gym . free weights , many machines , treadmills , etc . (...) *awesome location , 1 minute walk to bangla road and new shopping mall with movie theater . 1 minute walk to 2 starbucks* (…) great value for a low price .
Headline 2: *excellent location* , gym , sauna , pool . delicious food at the restaurant
Review 3: this is a solid 5 star hotel . *location is the best you can get . 1 minute walk to the beautiful modern shopping mall . 2 minute walk to the outdoor shopping and food market .* the hotel is new , *very clean* , excellent security to remove any worries . (...) i will stay here every trip to udon thani because of the *excellent location ,* low cost , *clean room .*
Headline 3: *great location . very clean .* very low price
Review 4: this was a very nice 3 star hotel . *great location across the street from the beach . 5 minute walk to walking street . 5 minute walk to the new mall .* (...) *room was perfect clean .*
Headline 4: *great location* , *clean hotel* , friendly staff , next to the beach

---

Review-headline pairs (in the test set) post by User-A:
Review : *very clean hotel* . rooms are like new . nice showers . beds are comfortable . *great location . just 2 minute walk to the bars . 10 minute walk to the big modern shopping mall* . staff speaks english . very low price for a new hotel .
Headline written by User-A: *very clean . great location*
Headline generated by seq2seq model: new hotel, very low price
Headline generated by seq2seq + UE + UD model: *very clean rooms . great location*

Fig. 2. From the review-headline pairs in the training set, we can figure out that User-A concerns more about the location (in red and *italic*) and cleanliness (in green and *italic*) than other respects for the hotel. Our **seq2seq + UE + UD** model taking advantage of user information successfully predicts the preference for User-A, while **seq2seq** model fails.

References

1. Amir, S., Wallace, B.C., Lyu, H., Carvalho, P., Silva, M.J.: Modelling context with user embeddings for sarcasm detection in social media. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning. pp. 167–177 (2016)
2. Carenini, G., Cheung, J.C.K., Pauls, A.: Multi-document summarization of evaluative text. Computational Intelligence pp. 545–576 (2013)
3. Chen, L., Qian, T., Zhu, P., You, Z.: Learning user embedding representation for gender prediction. In: Tools with Artificial Intelligence (ICTAI), 2016 IEEE 28th International Conference on. pp. 263–269 (2016)
4. Chen, W., Zhang, Z., Li, Z., Zhang, M.: Distributed representations for building profiles of users and items from text reviews. In: Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers. pp. 2143–2153 (2016)
5. Cho, K., van Merrienboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder–decoder approaches. In: Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. pp. 103–111 (2014)
6. Chopra, S., Auli, M., Rush, A.M.: Abstractive sentence summarization with attentive recurrent neural networks. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 93–98 (2016)
7. Gerani, S., Mehdad, Y., Carenini, G., Ng, R.T., Nejat, B.: Abstractive summarization of product reviews using discourse structure. In: Conference on Empirical Methods in Natural Language Processing. pp. 1602–1613 (2014)
8. Gu, J., Lu, Z., Li, H., Li, V.O.: Incorporating copying mechanism in sequence-to-sequence learning. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. pp. 1631–1640 (2016)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation pp. 1735–1780 (1997)
10. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, Usa, August. pp. 168–177 (2004)
11. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
12. Li, H., Zhu, J., Zhang, J., Zong, C.: Ensure the correctness of the summary: Incorporate entailment knowledge into abstractive sentence summarization. In: Proceedings of the 27th International Conference on Computational Linguistics (2018)
13. Li, J., Ritter, A., Hovy, E.: Weakly supervised user profile extraction from twitter. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. pp. 165–174 (2014)
14. Li, P., Lam, W., Bing, L., Guo, W., Li, H.: Cascaded attention based unsupervised information distillation for compressive summarization. In: Conference on Empirical Methods in Natural Language Processing. pp. 2081–2090 (2017)
15. Li, P., Lam, W., Bing, L., Wang, Z.: Deep recurrent generative decoder for abstractive text summarization. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 2091–2100 (2017)
16. Li, Z., Huang, J., Zhong, N.: Exploiting user and item embedding in latent factor models for recommendations. In: Proceedings of the International Conference on Web Intelligence. pp. 1241–1245 (2017)

17. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Proceedings of the Workshop on Text Summarization Branches Out. pp. 74–81 (2004)
18. Nallapati, R., Zhou, B., dos Santos, C., Gulcehre, C., Xiang, B.: Abstractive text summarization using sequence-to-sequence rnns and beyond. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning. pp. 280–290 (2016)
19. Nguyen, D.Q., Vu, T., Nguyen, T.D., Phung, D.: A capsule network-based embedding model for search personalization. arXiv preprint arXiv:1804.04266 (2018)
20. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: International Conference on Machine Learning. pp. 1310–1318 (2013)
21. Poussevin, M., Guigue, V., Gallinari, P.: Extended recommendation framework: Generating the text of a user review as a personalized summary. arXiv preprint arXiv:1412.5448 (2014)
22. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 379–389 (2015)
23. See, A., Liu, P.J., Manning, C.D.: Get to the point: Summarization with pointer-generator networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. pp. 1073–1083 (2017)
24. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research pp. 1929–1958 (2014)
25. Takase, S., Suzuki, J., Okazaki, N., Hirao, T., Nagata, M.: Neural headline generation on abstract meaning representation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1054–1059 (2016)
26. Vu, T., Nguyen, D.Q., Johnson, M., Song, D., Willis, A.: Search personalization with embeddings. In: European Conference on Information Retrieval. pp. 598–604 (2017)
27. Wang, L., Ling, W.: Neural network-based abstract generation for opinions and arguments. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 47–57 (2016)
28. Xu, S., Yang, S., Lau, F.: Keyword extraction and headline generation using novel word features. In: Twenty-Fourth AAAI Conference on Artificial Intelligence. pp. 1461–1466 (2010)
29. Yu, Y., Wan, X., Zhou, X.: User embedding for scholarly microblog recommendation. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. pp. 449–453 (2016)
30. Zhou, Q., Yang, N., Wei, F., Zhou, M.: Selective encoding for abstractive sentence summarization. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. pp. 1095–1104 (2017)