Towards Personalized Review Summarization via User-Aware Sequence Network

Junjie Li,[1,2] **Haoran Li,**[1,2] **Chengqing Zong**[1,2,3]
[1]National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]CAS Center for Excellence in Brain Science and Intelligence Technology, Beijing, China
{junjie.li, haoran.li, cqzong}@nlpr.ia.ac.cn

Abstract

We address personalized review summarization, which generates a condensed summary for a user's review, accounting for his preference on different aspects or his writing style. We propose a novel personalized review summarization model named User-aware Sequence Network (USN) to consider the aforementioned users' characteristics when generating summaries, which contains a user-aware encoder and a user-aware decoder. Specifically, the user-aware encoder adopts a user-based selective mechanism to select the important information of a review, and the user-aware decoder incorporates user characteristic and user-specific word-using habits into word prediction process to generate personalized summaries. To validate our model, we collected a new dataset *Trip*, comprising 536,255 reviews from 19,400 users. With quantitative and human evaluation, we show that USN achieves state-of-the-art performance on personalized review summarization.

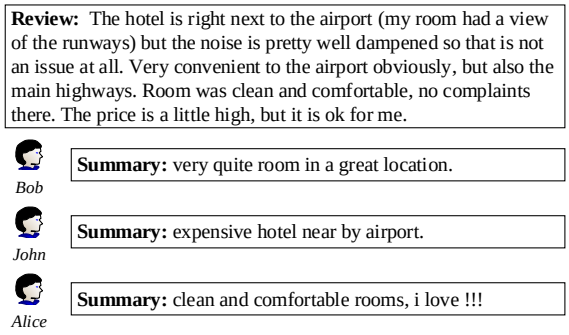

Figure 1: Personalized review summarization is motivated by that different users are likely to generate different summaries for the same review, according to their own experiences, thoughts, or writing styles.

Introduction

Review summarization aims to generate a condensed summary for a review or multiple reviews (Hu and Liu 2004; Ganesan 2010; Carenini, Cheung, and Pauls 2013; Gerani et al. 2014; Lu and Wang 2016). As this task can alleviate the information overload problem, it has been widely studied.

This paper addresses *personalization* issues of review summarization[1], which have not been discussed in previous research. Given a review, different users may care about different contents according to their own experiences or thoughts. Figure 1 illustrates the motivation with a hotel review sample. *Bob* may travel on business and he cares about ***location*** and ***room*** more than ***price***, while *John* may travel on a tight budget and care about ***price*** more. What's more, different users have their own writing styles. *Alice* often summarizes reviews with the words which can explicitly express her emotions, such as "love" or "hate", while *Bob* and *John* don't do that.

Actually, personalized review summarization is applicable to a wide range of online consumer review platforms, such as Tripadvisor and Yelp. For example, users often write reviews about their attitudes on hotels or restaurants in these websites. One function is to automatically generate summaries for these reviews using their own words and considering their preference on different aspects[2].

Another example would be helpful for users who read these reviews to choose products. Classical review summarization systems only summarize reviews based on review contents and show the same summaries to all readers. However, personalized review summarization can consider readers' preference and generate different summaries of reviews for different readers. These summaries can directly reflect aspects that readers care about and may be more suitable for them to choose products.

To perform personalized review summarization, we propose a User-aware Sequence Network (USN). USN is based on sequence to sequence models (*Seq2Seq*), which are popular methods in machine translation (Bahdanau, Cho, and Bengio 2015; Zhao et al. 2018), text summarization (Rush, Chopra, and Weston 2015; See, Liu, and Manning 2017; Li et al. 2018) and review summarization (Lu and Wang 2016;

[1]In this paper, we focus on single-review summarization and we leave adapting our model to multi-review summarization scenario to future work.

[2]aspects refer to a properties (attributes) of products or services, such as ***location*** and ***room*** for the hotel domain.

Ma et al. 2018). Our major updates over standard *Seq2Seq* are three-fold.

First, we design two kinds of user-based representations for personalized review summarization. One is user embedding, which embeds each user into a low-dimension vector and another is user-specific vocabulary memory, which stores user's active vocabulary.

Second, to consider users' different preferences on review content, we propose a *User-aware Encoder*, which utilizes a bidirectional-LSTM to encode review, and then it adopts a user-based selective mechanism to select the important information of the review to obtain a better representation.

Third, different from the classical decoder module in *Seq2Seq*, we propose a *User-aware Decoder* to consider different writing styles of users. It incorporates user embedding and user-specific vocabulary memory into word prediction module to generate personalized summaries.

To validate our approach, we collect a new personalized review summarization dataset named *Trip* from Tripadvisor website, which contains 536,255 review-summary pairs with 19,400 users. With quantitative and human evaluation, we show that USN achieves state-of-the-art performance on personalized review summarization. Our contributions are as follows:

- To the best of our knowledge, we first propose a user-aware *Seq2Seq*-based model named User-aware Sequence Network (USN) for personalized review summarization.
- Our model adopts a user-based selective mechanism considering different user preferences on review content when summarizing a review, and applies user-specific vocabulary to consider user's writing styles when generating a summary.
- For evaluation of personalized review summarization, we introduce a novel dataset named *Trip*, which is available at https://github.com/Junjieli0704/USN.

Background

In this section, we formalize our problem and introduce a classical sequence-to-sequence attentional model for review summarization.

Problem Formulation

Suppose we have a corpus D with m user-review-summary triples, and each triple contains a review x, a summary y and a user u who posts x and summarizes x to y. Review x consists of n words as $\{x_1, x_2, ..., x_n\}$, where $x_i \in V_s$ and V_s is the source vocabulary. Summary y consists of $l \leq n$ words as $\{y_1, y_2, ..., y_l\}$, where $y_i \in V_t$ and V_t is the target vocabulary. Personalized review summarization aims to generate summary y from review x by attending to u's characteristics on summarizing reviews.

Sequence-to-sequence attentional model

Sequence-to-sequence attentional model has been widely used in abstractive text and review summarization, and it contains two basic modules: encoder and decoder.

Encoder. Given review x, it first embeds each word x_i into vector $\mathbf{x}_i$ using embedding matrix $\mathbf{E}_v$. Then, these word vectors are fed into the encoder module (a single-layer bidirectional LSTM) one by one, producing a sequence of encoder hidden states $\mathbf{h}_i$.

Decoder. At each decoding time step t, the decoder (a single-layer unidirectional LSTM) receives previous word embedding to obtain the new hidden state $\mathbf{s}_t$. Then it computes context vector $\mathbf{c}_t$ for time step t through the attention mechanism:

$$\mathbf{e}_{t,i} = \mathbf{v}^T \tanh(\mathbf{W}_a \mathbf{h}_i + \mathbf{W}'_a \mathbf{s}_t + \mathbf{b}_a) \quad (1)$$

$$\alpha_{t,i} = \frac{\exp(\mathbf{e}_{t,i})}{\sum_i \exp(\mathbf{e}_{t,i})} \quad (2)$$

$$\mathbf{c}_t = \sum_i \alpha_{t,i} \mathbf{h}_i \quad (3)$$

where $\mathbf{W}_a, \mathbf{W}'_a, \mathbf{b}_a$ and $\mathbf{v}$ are parameters in the attention layer and $\alpha_{t,i}$ matches the importance score between current decoder state $\mathbf{s}_t$ and the encoder hidden state $\mathbf{h}_i$.

It then combines the context vector $\mathbf{c}_t$ and the decoder state $\mathbf{s}_t$ to construct the readout state $\mathbf{r}_t$, that is fed through a linear layer to produce the vocabulary distribution P_{voc}:

$$\mathbf{r}_t = \mathbf{W}_r[\mathbf{c}_t; \mathbf{s}_t] + \mathbf{b}_r \quad (4)$$

$$P_{\text{voc}} = \text{softmax}(\mathbf{W}_o \mathbf{r}_t + \mathbf{b}_o) \quad (5)$$

where $\mathbf{W}_o, \mathbf{W}_b, \mathbf{b}_o, \mathbf{b}_b$ are learnable parameters, and $[;]$ is the concatenating operator.

Finally, negative log likelihood is used for computing loss and the overall loss for the whole sequence is:

$$L = -\frac{1}{l} \sum_{t=0}^{l} \log P_{\text{voc}}(y_t) \quad (6)$$

User-aware Sequence Network

It is obvious that different users may care about different content of a review and have different word-using habits. Therefore we encode user information into encoder and decoder modules to model these different characteristics to perform personalized review summarization.

Specifically, we consider user from two views as follows: (1) user embedding (we embed user u as vector $\mathbf{u}$ and add $\mathbf{u}$ into our models) , (2) user-specific vocabulary memory, which is composed of K most user-specific words $\{U_k\}_{k=1}^K$ from u's previous reviews and summaries. After embedding each word in $\{U_k\}_{k=1}^K$ into vector $\{\mathbf{U}_k\}_{k=1}^K$ using embedding matrix $\mathbf{E}_v$, we can get the user-specific vocabulary memory $\mathbf{U}$ for user u. To build $\{U_k\}_{k=1}^K$, we first merge all reviews and summaries posted by u into a document. Then we compute $tf\text{-}idf$ scores for each word appears in the document, and we finally select top-K words for u. Using *tf-idf* scores means we do not include too general terms that many users commonly use, because they are not helpful for considering u.

Here, we introduce 4 strategies to incorporate user embedding and user-specific vocabulary memory into two basic modules (*User-aware Encoder* and *User-aware Decoder*) of User-aware Sequence Network (depicted in Figure 2).

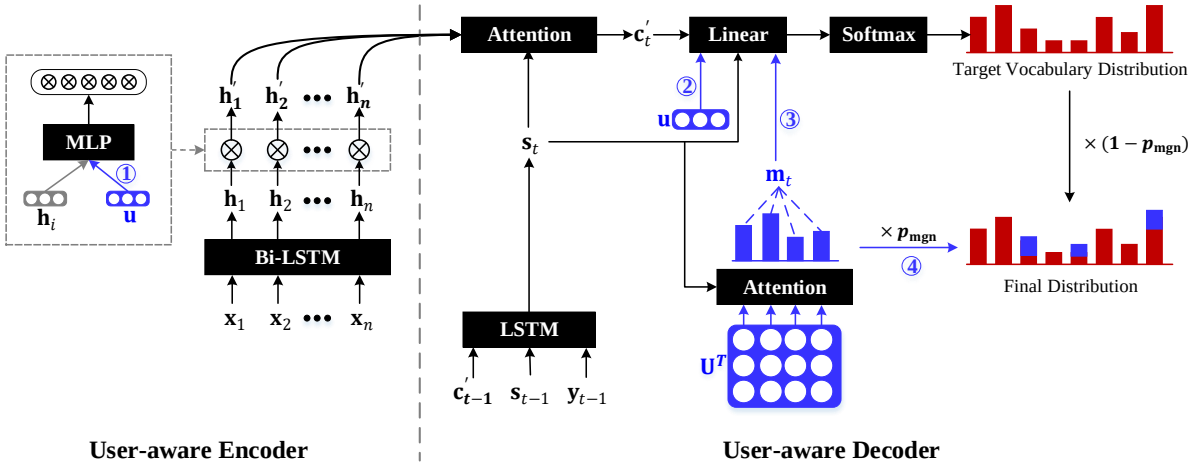

Figure 2: The architecture of User-aware Sequence Network (USN). USN encodes two kinds of user information, user embedding (**u**) and user-specific vocabulary memory (**U**), into its two basic modules (*User-aware Encoder* and *User-aware Decoder*). ①, and ② show strategies based on user embedding, and represent User Selection strategy, and User Prediction strategy, respectively. ③ and ④ indicate strategies based on user-specific vocabulary memory and represent User Memory Prediction strategy and User Memory Generation strategy, respectively.

User-aware Encoder

Different users pay attention to different content of a review. Inspired by (Zhou et al. 2017), we propose a user-based seletive mechanism to select important information from review for different users. The selective mechanism can construct a tailored representation of review x by considering u. In detail, our user-based seletive network takes user vector $\mathbf{u}$ and the encoder hidden state $\mathbf{h}_i$ as input, and outputs a gate vector $\mathbf{gate}_i$ to select $\mathbf{h}_i$.

$$\mathbf{gate}_i = \sigma(\mathbf{W}_k[\mathbf{h}_i; \mathbf{u}] + \mathbf{b}_k) \tag{7}$$

$$\mathbf{h}'_i = \mathbf{h}_i \odot \mathbf{gate}_i \tag{8}$$

where $\mathbf{W}_k$, $\mathbf{b}_k$ are learnable parameters, $[;]$ is the concatenating operator, σ denotes sigmoid activation function, and $\odot$ is element-wise multiplication.

From equation 7, we can find $\mathbf{gate}_i$ is a vector whose value is between 0 and 1. Therefore we can utilize $||\mathbf{gate}_i||_2$ to measure the degree of the filter and call it *2-Norm Gate Value*. High value means most of the information in $\mathbf{h}_i$ is passed from the filter, which results in the word $\mathbf{x}_i$ is important. This is the first strategy we proposed to take users into consideration called ***User Selection strategy***.

User-aware Decoder

After selecting important information using user-aware encoder, we obtain new hidden state $\mathbf{h}'_i$ for i-th word in review x. Classical attention module is also applied to compute context vector $\mathbf{c}'_t$ for time step t.

When generating a summary, different users may have their own vocabulary. Thus it is natural to take user-specific vocabulary memory into consideration when predicting output vocabulary distribution P_{voc} and different words in user-specific vocabulary may have different effects. Thus, we utilize an attention mechanism to extract important words in $\mathbf{U}$ when obtaining vocabulary state $\mathbf{m}_t$.

$$\mathbf{g}_{t,k} = \mathbf{v}^T \tanh(\mathbf{W}_m \mathbf{U}_k + \mathbf{W}'_m \mathbf{s}_t + \mathbf{b}_m) \tag{9}$$

$$\beta_{t,i} = \frac{\exp(\mathbf{g}_{t,k})}{\sum_i \exp(\mathbf{g}_{t,k})} \tag{10}$$

$$\mathbf{m}_t = \sum_k \beta_{t,k} \mathbf{U}_k \tag{11}$$

where $\mathbf{W}_m$, $\mathbf{W}'_m$ and $\mathbf{b}_m$ are parameters. Therefore, we can enhance the readout state $\mathbf{r}_t$ by considering $\mathbf{m}_t$. Besides, we can also enhance the readout state $\mathbf{r}_t$ by combining user vector $\mathbf{u}$.

$$\mathbf{r}'_t = \mathbf{W}_{r'}[\mathbf{c}'_t; \mathbf{s}_t; \mathbf{u}; \mathbf{m}_t] + \mathbf{b}_{r'} \tag{12}$$

$$P'_{\text{voc}} = \text{softmax}(\mathbf{W}_o \mathbf{r}'_t + \mathbf{b}_o) \tag{13}$$

where $\mathbf{W}_{r'}$ and $\mathbf{b}_{r'}$ are learnable valuables.

The strategies of adding user vector $\mathbf{u}$ and vocabulary state $\mathbf{m}_t$ into readout state $\mathbf{r}'_t$ are called ***User Prediction strategy*** and ***User Memory Prediction strategy***, respectively.

Last but not least, inspired by (See, Liu, and Manning 2017), we also propose a soft copy mechanism to copy user-specific words in generating summaries, which is the 4-*th* strategy called ***User Memory Generation strategy***.

The generation probability $p_{\text{mgn}} \in [0, 1]$ for timestep t is calculated from the context vector $\mathbf{c}'_t$, the decoder state $\mathbf{s}_t$ and the vocabulary state $\mathbf{m}_t$:

$$p_{\text{mgn}} = \sigma(\mathbf{W}_{mg}[\mathbf{c}'_t; \mathbf{s}_t; \mathbf{m}_t] + \mathbf{b}_{mg}) \tag{14}$$

where $\mathbf{W}_{mg}, \mathbf{b}_{mg}$ are learnable parameters, $[;]$ is the concatenating operator and σ is the sigmoid function. Next p_{mgn} is used as a soft switch to choose between generating a word

from the target vocabulary V_t or coping a word from user-specific vocabulary.

$$P(w) = (1 - p_{\text{mgn}})P'_{\text{voc}}(w) + p_{\text{mgn}} \sum_{k:U_k=w} \beta_{t,k} \quad (15)$$

The first part in Equation 15 represents generating words from our vocabulary, and the second part indicates coping words from user-specific vocabulary memory, respectively.

Dataset

Since there is no available personalized review summarization dataset, we create a new one named *Trip*. *Trip* is collected from Tripadvisor, which is a travel review website that contains user-generated reviews along with their authors and titles. The title of a review often summarizes the important information of the review, therefore we take the title as the reference summary of the review. We collect 2,832,874 user-review-summary triplets. However, when we observe these triplets carefully, we find there are many noisy samples. Since there is no any constraints on writing a title, users may write titles arbitrarily and it results in many meaningless titles, such as "not my first choice", "i will be back again", and "twice in one trip". To remove these noisy samples from our original dataset, we propose three filters:

(1) Aspect-based filter is used to remove samples whose title does not describe any aspects. We manually define 6 common aspects in the hotel domain (***location***, ***service***, ***room***, ***value***, ***facility*** and ***food***) as well as their seed words shown in table 2. Then *Aspect Segmentation* (Wang, Lu, and Zhai 2010) algorithm is used to expand their seed words automatically with boot-strapping strategy. Hyperparameters in *Aspect Segmentation*, such as selection threshold and iteration times, are set as the same with (Wang, Lu, and Zhai 2010). Finally, we remove samples that all words in review title do not appear in our seed words for all aspects.

(2) Title length filter is used to remove samples whose title are too short (length is less than 5).

(3) Compression ratio filter is used to remove samples whose ratio between the length of a review and a title is larger than 50.

After applying these filters, we construct *Trip* with 536,255 user-review-summary triplets. We also randomly choose 1,000 samples from *Trip* and check the aspects overlap between the reviews and the summaries by humans. We find that the percentage of all samples whose aspects described in a summary all appear in its corresponding review surpasses 90%, which reveals the reliability of our dataset.

Statistics of *Trip* are summarized in Table 1. We randomly split the dataset into 5,000 reviews for test, 5,000 reviews for validation and the rest for training. The input and output vocabularies are collected from the training data, which have 360,448 and 48,378 word types respectively.

Experiments

In this section we introduce the evaluation metric, all the comparison methods, the implementation details, and the performance of our model.

#reviews	536,255	#summaries	536,255
#users	19,400	#reviews/user	27.64
#words/review	154.79	#words/summary	7.60

Table 1: Data statistics for *Trip*.

Aspect	Keywords
location	location, traffic, minute, walk
service	server, service, welcome, staff
room	room, bed, clean, dirty
value	value, price, quality, worth
facility	pool, parking, internet, wifi
food	delicious, breakfast, coffee, cheese

Table 2: Aspects and their keywords for *Trip*.

Evaluation Metric

We exploit ROUGE (Lin 2004) as our evaluation metric. ROUGE scores reported in this paper are computed by Pyrouge package [3].

Comparison Methods

As far as we know, all previous review summarization studies focused on the multi-review summarization scenario, which is essentially different from our task. Here, we compare with several methods which are popular in text summarization and can be divided into two types: extractive and abstractive summarization approaches.

- *Lead*-1 is an extractive approach which selects the first sentence in review as summary.
- *LexRank* (Erkan and Radev 2004) is also an an famous extractive approach that computes text centrality based on PageRank algotithm.
- *S2S+Att* is sequence to sequence model with attention implemented by us.
- *SEASS* (Zhou et al. 2017) adopts a selective network to select important information from review into *S2S+Att* and obtains state-of-the-art results in sentence summarization.
- *PGN* (See, Liu, and Manning 2017) adopts a copy mechanism to copy words from review when generating summarization into *S2S+Att* and obtains state-of-the-art results in document summarization.

Implementation Details

For all experiments, we set the word embedding size and user embedding size to 128, and all LSTM hidden state sizes to 256. We use dropout (Srivastava et al. 2014) with probability p = 0.2. During training, we use loss on the validation set to implement early stopping and also apply gradient clipping (Pascanu, Mikolov, and Bengio 2013) with range $[-5, 5]$. At test time, our summaries are produced using beam search with beam size 5.

We use Adam as our optimizing algorithm. We set the batch size to 128. We use a vocabulary of 30,000 words for

[3] pypi.python.org/pypi/pyrouge/0.1.3

Models	RG-1	RG-2	RG-L
Lead-1	12.77	2.98	11.24
LexRank	10.84	1.88	9.46
S2S+Att	22.09	6.39	20.36
SEASS	21.77	6.14	20.12
PGN	22.51	6.89	20.79
USN	**24.78***	**7.75***	**22.66***

Table 3: ROUGE F1 scores on the test set. RG in the Table denotes ROUGE. Models and baselines in the top half are extractive methods, while those in the bottom half are abstractive ones. The best performance is in **bold**. The superscript * indicates our USN model performs significantly better than all baseline models as given by the 95% confidence interval in the official ROUGE script.

both source and target. We truncate the review to 200 tokens, which is done to expedite training and testing. However we also find that truncating the review can raise the performance of the model[4]. We use the develop set to choose the size of user-specific vocabulary and set it to 200.

Results

The results are displayed in Table 3. For extractive methods, we can see that *Lead*-1 performs best. However, it only obtains 12.77 ROUGE-1, 2.98 ROUGE-2 and 11.27 ROUGE-L F1 scores. The reason is that summaries in *Trip* are very succinct and often cover content across sentences. Therefore, the extractive methods perform poor. That is also the reason why we build our user-aware review summarization model based on abstractive methods.

For abstractive methods, we can find that *S2S+Att* is better than all extractive methods. After adding selective mechanism into *S2S+Att*, the performance of *SEASS* decreases slightly. The reason is that the selective mechanism proposed by *SEASS* is designed for sentence summarization, which may not be suit for summarizing documents (reviews). The average length of input is less than 40 in Zhou et al. 2017, while the average length in *Trip* is about 154. However, using selective mechanism to filter the input is very necessary for summarization. Therefore, we propose a user-based selective mechanism to filter the input and our method is proved to be effective. When considering copy mechanism into *S2S+Att*, *PGN* obtains better performance.

Finally, after incorporating our proposed 4 user-based strategies into *S2S+Att*, *USN* achieves 24.78 ROUGE-1, 7.75 ROUGE-2 and 22.66 ROUGE-L F1 scores and performs significantly better than all previous methods. Compared to *S2S+Att*, our model has a 2.69 ROUGE-1, 1.36 ROUGE-2 and 2.3 ROUGE-L gains, which shows explicitly modeling user-related characteristics can indeed improve summarization quality. Our model also surpasses *PGN* by 2.27 ROUGE-1, 0.86 ROUGE-2 and 1.87 ROUGE-L and achieves the state-of-the-art performance on review summarization.

[4]Indeed, we found that using only the first 200 tokens of the review yields higher ROUGE scores than using the first 600 tokens.

Models	Precision	Recall	F1
S2S+Att	0.516	0.502	0.509
PGN	0.518	0.542	0.530
USN	**0.587**	**0.601**	**0.594**

Table 4: Aspect-level Precision, Recall and F1 for different systems

Human Evaluation on Aspect-level Coverage

USN is a personalized model, which can not only capture word-level user preference, but also capture aspect-level user preference. Aspects that user care about often appear in gold summaries. Therefore, we want to identify whether aspects described in summaries generated by *USN* are consistent with aspects described in gold summaries.

We utilize 6 aspects (***location***, ***service***, ***room***, ***value***, ***facility*** and ***food***) provided by Table 2 as our gold aspects. Beyond that, we also add an aspect to describe the overall attitude and name it ***hotel***. Given a summary, we label it with the aforementioned 7 aspects. Two summaries with human labeled aspect are as follows:

Example. 1 friendly staff with good room (***service***, ***room***)

Example. 2 a great hotel in city center (***hotel***, ***location***)

To perform this human evaluation, we randomly sample 1000 user-review-summary triplets from our test set. We first generate summaries of these reviews from *S2S+Att*, *PGN* and *USN*. Then we ask two students to label aspects to these gold and generated summaries. After that, we compute aspect-level precision, recall and F1 for different systems and show it in Table 4. We can find that *USN* outperforms other models (*S2S+Att* and *PGN*) by a large margin, which shows our model can capture aspect-level user preference.

Discussions

Effects of Different User-based Strategies

In this paper, we propose 4 user-based strategies to construct our user-aware review summarization model. To evaluate the effect of each strategy on personalized review summarization, we perform the ablation study of *USN* and report experiment results in Table 6.

First, we find that models which adds only one kind of user-based strategy into *S2S+Att* (line 2-5) can obtain at least 0.83 in ROUGE-1 compared with *S2S+Att*. It shows that all these strategies improve the performance of personalized review summarization. User Prediction and User Memory Prediction strategies are the two most effective strategies, the reason is that they directly affect the word prediction module in *USN*.

Second, models which deletes one kind of user-based strategy from *USN* (line 6-9) will descend at least 0.14 in ROUGE-1 compared with *USN* (line 10). It shows all our four user-based strategies are complementary. The most complementray one is User Selection strategy, which is applied on the encoder module of *USN*, while others are applied on the decoder module of *USN*.

Review from *UserA*:	i was here one night for a business meeting in the hotel . i was immediately impressed with the front-desk staff when checking in . i had several needs , and they were very accommodating . the bed was perfect - super comfortable . the restaurant food is quite decent , as well . i definitely recommend this hotel and would return .

Table 5: User-based selective gate visualization of a input review. The important words are selected from the input review, such as "impressed", "staff", "bed" and "perfect". The output summary of our model is"excellent service , comfy be" and the true summary is "excellent service , very comfortable bed".

line	USel	UPre	UMP	UMG	RG-1	RG-2	RG-L
1	–	–	–	–	21.15	6.08	19.56
2	✓	–	–	–	22.29	6.20	20.41
3	–	✓	–	–	24.04	7.30	21.92
4	–	–	✓	–	23.36	6.84	21.40
5	–	–	–	✓	21.98	6.49	20.33
6	–	✓	✓	✓	23.23	6.64	21.17
7	✓	–	✓	✓	24.64	7.88	22.56
8	✓	✓	–	✓	24.11	7.43	22.22
9	✓	✓	✓	–	24.52	7.56	22.28
10	✓	✓	✓	✓	24.78	7.75	22.66

Table 6: Effects of different user-based strategies on review summarization. USel, UPre, UMP and UMG denote User Selection strategy, User Prediction strategy, User Memory Prediction strategy and User Memory Generation strategy, respectively. "✓" means our model considers the specific strategy, while "–" means not. When there is no user-based strategies considered in our model, our model degrades into *S2S+Att* (line 1).

Third, after merging all strategies into *S2S+Att*, *USN* obtains the best result.

Effects of user-specific vocabulary size

Since User Memory Prediction (*UMP*) and User Memory Generation (*UMG*) strategies are based on user-specific vocabulary, we show the effect of user-specific vocabulary size of these two strategies on development set of *Trip* in Figure 3, our observations are follows:

First, we can see that adopting user-specific vocabulary into *S2S+Att* with *UMP* or *UMG* can indeed improve the performance of review summarization, even though when the user-specific vocabulary size is small (such as 50). Second, by comparing the ROUGE-1 F1 Scores between *UMG* and *UMP*, we can find that *UMP* obtains higher performance, the reason is that *UMP* can directly affect the word prediction module. Finally, we set our user-vocabulary size to 200, since *S2S+Att+UMP+UMG* obtains the best performance at the point.

User-based Selective Gate Visualization

In order to validate that our model is able to select important words in a review for personalized review summarization, we visualize user-based selective gate value in Table 5.

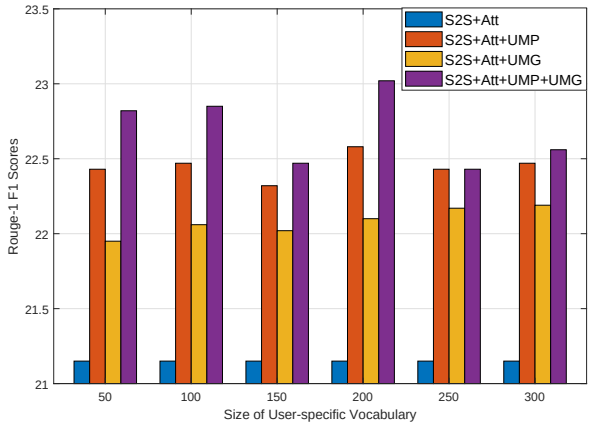

Figure 3: Effects of user-specific vocabulary size on development set of *Trip*.

Word with dark color means high *2-Norm Gate Value* (see the section of User-aware Encoder) and results in a important word. From the gold summary given by *UserA*, we can find *UserA* may care about "service" and "bed" more and the important words found by our user-based selective mechanism are "impressed", "staff", "bed" and "perfect", which also reflects *UserA*'s experience on these two aspects. It shows our personalized model can mine the important information for users.

Case Study

We show the case study of a sample from *Trip* test set in Figure 4.

First, although the review describes *UserB*'s attitudes on ***room***, ***food***, ***service***, and ***location***, the reference only contains ***room*** and ***location***. This shows *UserB* cares these two aspects more. Actually, we observe all reviews posted by *UserB*. There are 40 reviews with summaries posted by *UserB*, more than 80% these reviews and summaries contain *UserB*'s attitude on these two aspects. Existing methods without modeling user information (*S2S+Att* and *PGN*) cannot capture *UserB*'s preference on these two aspects, which results in these methods generate words (such as the "staff") about ***service***. While our personalized model can mine such preference and only generate words about ***location*** and ***room***.

Second, the word "comfortable" is hard to generate, because it does not appear in the review. However, we find that

Review from *UserB*: i stayed here for three nights and i felt really pleasing. it had a fully equipped kitchen, a lounge and the shower was great too. the bar downstairs gets pretty busy, but i could not hear much from our room. the staff was always friendly and helpful. they offer a decent breakfast. it is opposite the train station, but i was on my bike so i found it easy to get to nearby attractions ...	
S2S+Att:	nice hotel with good room
PGN:	friendly staff with good room
USN:	great location , ***comfortable*** room
Gold:	great location with ***comfortable*** room

Figure 4: An example of the generated review summairzation of *S2S+Att*, *PGN* and *USN* (Italic and bold denote words that do not appear in review).

it appears in *UserB*-specific vocabulary. After merging the vocabulary, *USN* can generate the word accurately.

Related Work

Review summarization belongs to sentiment analysis (Liu 2016; Xia et al. 2015), which is a large area in natural language processing and contains sentiment classification (Li and Zong 2008; Xia, Zong, and Li 2011; Li, Yang, and Zong 2016; 2018), emotion dection (Li et al. 2015b), spam detection (Wang, Liu, and Zhao 2017) and so on. Personalized review summarization is related to both opinion summarization and personalized text summarization.

Opinion summarization

There are two mainstream approaches for opinion summarization: extractive and abstractive approaches. A key task in extractive methods (Hu and Liu 2004; Lerman, Blair-Goldensohn, and Mcdonald 2009; Xiong and Litman 2014) is to identify important text units. For example, Hu and Liu (2004) first recognize the frequent product features and then attach extracted opinion sentences to the corresponding feature. Xiong and Litman (2014) exploit review helpfulness for review summarization. However, many studies (Carenini, Cheung, and Pauls 2013; Fabbrizio, Stent, and Gaizauskas 2014) have shown that abstractive approaches may be more appropriate for summarizing evaluative text than extractive ones. Therefore, in this paper, we study abstract generation techniques to summarize reviews.

Abstractive approaches (Ganesan 2010; Carenini, Cheung, and Pauls 2013; Fabbrizio, Stent, and Gaizauskas 2014; Gerani et al. 2014; Lu and Wang 2016) are also very popular methods in review summarization. For example, Ganesan (2010) first represent review as token-based graphs based on the token order in the string and then rank summary candidates by scoring paths after removing redundant information from the graph. Gerani et al. (2014) utilize discourse structure of review to identify important aspects and then design a set of templates to generate summarizations. Lu and Wang (2016) propose an attention-based neural network model for generating abstractive summaries of opinionated text.

All these studies focus on review summarization in the multiple review scenario, while our work focuses on personalization issues in single review summarization scenario. Ma et al. (2018) is also related to our work, which jointly models review summarization and sentiment classification in an unified framework. However, this work also ignores the effect of users on review summarization, while our task is personalized review summaization and our model can consider the effect of users on review summarization.

Personalized text summarization

Personalized text summarization is an active area (Zhang et al. 2003; Yan, Nie, and Li 2011; Díaz and Gervás 2007; Yang et al. 2012; Móro and Bielikov 2012; Li et al. 2015a). These studies either employ interactive user clicks/examinations (Yan, Nie, and Li 2011) or utilize users' annotations or highlights (Zhang et al. 2003; Móro and Bielikov 2012) to capture user preference and perform personalized summarization. Although these studies boost the performance of text summarization, their user information is hard to obtain. Different with them, we only utilize users' previous reviews to mine user preference, which is very easy to obtain. Another difference between our work and these studies is that they all focus on news-based text summarization, while we focus on review summarization.

Poussevin, Guigue, and Gallinari (2015) and Wang and Zhang (2017) also summarize review in a personal situation to boost recommendation system. However, our work and these two studies are not in the same scenario. To generate a personalized summary of product p for user u, they need a set of reviews posted by u and a set of reviews about p. Our work is to summarize a review posted by u by considering u's characteristics.

Conclusion and Future Work

In this paper, we address personalized issue of review summarization and propose a User-aware Sequence Network (USN) to consider user information into personalized review summarization. USN contains two basic modules: user-aware encoder and user-aware decoder. To build these two user-aware modules, we propose 4 user-based strategies. To validate our model, we construct a new dataset (*Trip*). Extensive experiments on *Trip* show that USN outperforms state-of-the-art methods significantly.

This paper focuses on personalized review summarization in the single-review scenario. For future work, we wish to extend our model to the multi-review scenario.

Acknowledgements

We thank Junnan Zhu and Xiaomian Kang for valuable discussions. The research work descried in this paper has been supported by the National Key Research and Development Program of China under Grant No. 2016QY02D0303.

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Carenini, G.; Cheung, J. C. K.; and Pauls, A. 2013. Multi-document summarization of evaluative text. *Computational Intelligence* 29(4):545–576.

Díaz, A., and Gervás, P. 2007. User-model based personalized summarization. *Information Processing and Management* 43(6):1715–1734.

Erkan, G., and Radev, D. R. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.* 22:457–479.

Fabbrizio, G. D.; Stent, A.; and Gaizauskas, R. 2014. A hybrid approach to multi-document summarization of opinions in reviews. In *INLG*, 54–63.

Ganesan, K. 2010. Opinosis:a graph-based approach to abstractive summarization of highly redundant opinions. In *COLING*, 340–348.

Gerani, S.; Mehdad, Y.; Carenini, G.; Ng, R. T.; and Nejat, B. 2014. Abstractive summarization of product reviews using discourse structure. In *EMNLP*, 1602–1613.

Hu, M., and Liu, B. 2004. Mining and summarizing customer reviews. In *SIGKDD*, 168–177.

Lerman, K.; Blair-Goldensohn, S.; and Mcdonald, R. 2009. Sentiment summarization: evaluating and learning user preferences. In *EACL*, 514–522.

Li, S., and Zong, C. 2008. Multi-domain sentiment classification. In *ACL*, 257–260.

Li, P.; Bing, L.; Lam, W.; Li, H.; and Liao, Y. 2015a. Reader-aware multi-document summarization via sparse coding. In *IJCAI*, 1270–1276.

Li, S.; Huang, L.; Wang, R.; and Zhou, G. 2015b. Sentence-level emotion classification with label and context dependence. In *ACL*, 1045–1053.

Li, H.; Zhu, J.; Zhang, J.; and Zong, C. 2018. Ensure the correctness of the summary: Incorporate entailment knowledge into abstractive sentence summarization. In *COLING*, 1430–1441.

Li, J.; Yang, H.; and Zong, C. 2016. Sentiment classification of social media text considering user attributes. In *NLPCC*, 583–594.

Li, J.; Yang, H.; and Zong, C. 2018. Document-level multi-aspect sentiment classification by jointly modeling users, aspects, and overall ratings. In *COLING*, 925–936.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.

Liu, B. 2016. *Sentiment analysis: Mining opinions, sentiments, and emotions*. University of Cambridge: Cambridge University Press.

Lu, W., and Wang, L. 2016. Neural network-based abstract generation for opinions and arguments. In *NAACL*, 47–57.

Ma, S.; Sun, X.; Lin, J.; and Ren, X. 2018. A hierarchical end-to-end model for jointly improving text summarization and sentiment classification. In *IJCAI*, 4251–4257.

Móro, R., and Bielikov, M. 2012. Personalized text summarization based on important terms identification. In *International Workshop on Database and Expert Systems Applications*, 131–135.

Pascanu, R.; Mikolov, T.; and Bengio, Y. 2013. On the difficulty of training recurrent neural networks. In *ICML*, 1310–1318.

Poussevin, M.; Guigue, V.; and Gallinari, P. 2015. Extended recommendation framework: Generating the text of a user review as a personalized summary. In *RecSys*, 34–41.

Rush, A. M.; Chopra, S.; and Weston, J. 2015. A neural attention model for abstractive sentence summarization. In *EMNLP*, 379–389.

See, A.; Liu, P. J.; and Manning, C. D. 2017. Get to the point: Summarization with pointer-generator networks. In *ACL*, 1073–1083.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958.

Wang, Z., and Zhang, Y. 2017. Opinion recommendation using a neural model. In *EMNLP*, 1626–1637.

Wang, X.; Liu, K.; and Zhao, J. 2017. Handling cold-start problem in review spam detection by jointly embedding texts and behaviors. In *ACL*, 366–376.

Wang, H.; Lu, Y.; and Zhai, C. 2010. Latent aspect rating analysis on review text data: a rating regression approach. In *SIGKDD*, 783–792.

Xia, R.; Xu, F.; Zong, C.; Li, Q.; Qi, Y.; and Li, T. 2015. Dual sentiment analysis: Considering two sides of one review. *IEEE Trans. Knowl. Data Eng.* 27(8):2120–2133.

Xia, R.; Zong, C.; and Li, S. 2011. Ensemble of feature sets and classification algorithms for sentiment classification. *Information Science* 181(6):1138–1152.

Xiong, W., and Litman, D. 2014. Empirical analysis of exploiting review helpfulness for extractive summarization of online reviews. *Grantee Submission*.

Yan, R.; Nie, J. Y.; and Li, X. 2011. Summarize what you are interested in: An optimization framework for interactive personalized summarization. In *EMNLP*, 1342–1351.

Yang, G.; Wen, D.; Chen, N. S.; and Sutinen, E. 2012. Personalized text content summarizer for mobile learning: An automatic text summarization system with relevance based language model. In *IEEE Fourth International Conference on Technology for Education*, 90–97.

Zhang, H.; Chen, Z.; Ma, W. Y.; and Cai, Q. 2003. A study for documents summarization based on personal annotation. In *HLT-NAACL*.

Zhao, Y.; Zhang, J.; He, Z.; Zong, C.; and Wu, H. 2018. Addressing troublesome words in neural machine translation. In *EMNLP*, 391–400.

Zhou, Q.; Yang, N.; Wei, F.; and Zhou, M. 2017. Selective encoding for abstractive sentence summarization. In *ACL*, 1095–1104.