Neural Computing and Applications (2020) 32:6421–6433
https://doi.org/10.1007/s00521-018-3825-2

MULTI-SOURCE DATA UNDERSTANDING (MSDU)

Cross-domain aspect/sentiment-aware abstractive review summarization by combining topic modeling and deep reinforcement learning

Min Yang[1] · Qiang Qu[1] · Ying Shen[2] · Kai Lei[3] · Jia Zhu[4]

Received: 25 July 2018 / Accepted: 15 October 2018 / Published online: 31 October 2018

Abstract

Review text has been widely studied in traditional tasks such as sentiment analysis and aspect extraction. However, to date, no work is toward the end-to-end abstractive review summarization that is essential for business organizations and individual consumers to make informed decisions. This study takes the lead to study the aspect/sentiment-aware abstractive review summarization in *domain adaptation scenario*. Our novel model Abstractive review Summarization with Topic modeling and Reinforcement deep learning (ASTR) leverages the benefits of the supervised deep neural networks, reinforcement learning, and unsupervised probabilistic generative model to strengthen the aspect/sentiment-aware review representation learning. ASTR is a multi-task learning system, which simultaneously optimizes two coupled objectives: domain classification (auxiliary task) and abstractive review summarization (primary task), in which a document modeling module is shared across tasks. The main purpose of our multi-task model is to strengthen the representation learning of documents and safeguard the performance of cross-domain abstractive review summarization. Specifically, ASTR consists of two key components: (1) a domain classifier, working on datasets of both source and target domains to recognize the domain information of texts and transfer knowledge from the source domain to the target domain. In particular, we propose a weakly supervised LDA model to learn the domain-specific *aspect* and *sentiment lexicon* representations, which are then fed into the neural hidden states of given reviews to form aspect/sentiment-aware review representations; (2) an abstractive review summarizer, sharing the document modeling module with the domain classifier. The learned aspect/lexicon-aware review representations are fed into a pointer-generator network to generate aspect/sentiment-aware abstractive summaries of given reviews by employing a reinforcement learning algorithm. We conduct extensive experiments on real-life Amazon reviews to evaluate the effectiveness of our model. Quantitatively, ASTR achieves better performance than the state-of-the-art summarization methods in terms of ROUGE score and human evaluation in both out-of-domain and in-domain setups. Qualitatively, our model can generate better sentiment-aware summarization for reviews with different categories and aspects.

Keywords Domain adaptation · Abstractive review summarization · Reinforcement learning · Weakly supervised LDA

1 Introduction

User-generated reviews on products are expanding rapidly with the emergence of e-commerce. These reviews are valuable to business organizations for improving their products and to individual consumers for making informed decisions. Unfortunately, reading through all the product reviews is hard, especially for the reviews that are long and have low readability. It is therefore essential to provide coherent and concise summaries of user-generated reviews. In this paper, we focus on generating abstractive

✉ Qiang Qu
qiang@siat.ac.cn

Min Yang
min.yang@siat.ac.cn

Ying Shen
shenying@pkusz.edu.cn

Kai Lei
leik@pkusz.edu.cn

Jia Zhu
jzhu@m.scnu.edu.cn

Extended author information available on the last page of the article

summaries of product reviews in *domain adaptation scenario*. Abstractive text summarization is the task of generating a short and concise summary that captures the salient ideas of the source text. The generated summaries potentially contain new phrases and sentences that may not appear in the source text. Inspired by the recent success of sequence-to-sequence (seq2seq) model in statistical machine translation, most abstractive summarization systems employ seq2seq framework to generate summaries [29, 32, 37]. In general, the seq2seq model firstly uses an encoder to convert the input text into a vector representation, and then, it feeds this representation into a decoder to generate the summary.

Despite the remarkable progress of previous studies, generating aspect/sentiment-aware summaries of product reviews remains a challenge in real-world for three reasons. (1) First, neural sequence-to-sequence models tend to generate the trivial and generic summary, often involving high-frequency phrases. These summaries cannot capture the aspect and sentiment information from the product reviews which play a vital role in helping customers to make quick and informed decisions on certain products [21]. (2) Prior work trains the summarization model with a large number of labeled data. Nevertheless, annotating sufficient data is labor-intensive and time-consuming, establishing significant barriers to adapting the summarization systems to new domains which have limited labeled data. (3) According to what we observe, summary styles and words in different categories can significantly vary. However, existing methods apply a uniform model to generate text summaries for the source documents in different categories, which easily miss or under-represent salient aspects of the documents. (4) The decoder of the seq2seq model often takes as input the word embedding of the previous ground truth word during the training. However, at test time, the decoder takes the previous word emitted by the model as input. This will lead to the exposure bias problem [35] at the testing phase, which may accumulate errors quickly at each time step. Concretely, once the decoder produces a “bad” word, the model will propagate and accumulate errors with the increase in the length of the generated sequence.

To alleviate these limitations, we propose a novel “ASTR” model, which leverages the benefits of the supervised deep neural networks, reinforcement learning as well as the unsupervised probabilistic generative model for cross-domain abstractive review summarization. ASTR is a multi-task system, in which a document modeling module is shared across tasks. The main purpose of our multi-task model is to make the learned sentence representation be aware of different domains and contribute to the knowledge transfer from source domains to target domains. A domain classifier, the first subtask in ASTR, is trained for the reviews in both source and target domains. In particular, a weakly supervised LDA model (wsLDA) is proposed to learn domain-specific aspect and sentiment lexicon representations which are then used to calculate the aspect/sentiment-aware review representations via a multi-view attention mechanism. The domain classifier helps the semantic analysis and comprehension of reviews, which serves as the prior information for cross-domain aspect/sentiment-aware abstractive review summarization. It is expected to contribute to the knowledge transfer from source domains to target domains. The abstractive review summarizer, the second subtask in ASTR, shares the document modeling module with the domain classifier. The learned aspect/lexicon-aware review representations are fed into a pointer-generator network to generate aspect/sentiment-aware abstractive summaries of given reviews by employing the reinforcement learning algorithm. Similar to [37], we use an attention-based LSTM encoder–decoder architecture as the backbone of abstractive review summarizer to generate the abstractive summary.

We summarize our main contributions as follows:

- To the best of our knowledge, this is the first work dealing with abstractive review summarization in the domain adaptation scenario. In addition, it takes the lead to study the aspect/sentiment-aware abstractive review summarization in an end-to-end manner without hand-crafted features and templates by exploring the encoder–decoder framework.
- We propose ASTR, which leverages domain classification task to learn better review representations and transfer knowledge from the source domain to the target domain.
- ASTR integrates the supervised deep learning system with the unsupervised probabilistic generative model to strengthen the representation learning via an attention mechanism. The learned representation is expected to capture the aspect and sentiment knowledge.
- As the aspect and sentiment information is crucial to abstractive review summarization, we propose a weakly supervised LDA (wsLDA) model to automatically learn domain-specific aspect and sentiment representations from both source and target data. Furthermore, a multi-view attention mechanism is designed to learn aspect/sentiment-aware review representations, which captures the important information from different representation subspaces at different positions.
- The comprehensive experiments show that our model outperforms the competitors from both quantitative and qualitative perspectives.

The remainder of this paper is organized as follows. Section 2 reviews and discusses the related work, including abstractive text summarization, opinion summarization, multi-task learning, and domain adaptation. In Sect. 3, we fully introduce the proposed ASTR model. The experimental setup is introduced in Sect. 4, including the evaluation datasets, the compared methods, and the implementation details. Section 5 shows the quantitative and qualitative evaluation results and analysis. Section 6 makes the conclusions and discusses the future work.

2 Related work

2.1 Abstractive text summarization

In general, existing text summarization approaches can be categorized as extractive and abstractive. The extractive summarization copies representative sentences from the input [48], while the abstractive summarization generates new phrases, possibly rephrasing or using words that are not in the original text [36]. In this paper, we focus on abstractive text summarization systems.

Inspired by the recent success of the encoder–decoder framework in statistical machine translation, there has been increasing interest in generalizing the neural language model to the field of abstractive summarization [3, 6, 29, 36, 37]. For example, Rush et al. [36] were the first to apply the attention-based encoder–decoder model to abstractive text summarization, achieving state-of-the-art performance two sentence-level summarization datasets. Nallapati et al. [29] proposed off-the-shelf attention encoder–decoder RNN that captured the hierarchical document structure and identified the key sentences and keywords in the document. See et al. [37] proposed a hybrid pointer-generator network that allowed both copying words from the source text via pointing and generating words from a fixed vocabulary.

Several recent studies attempted to integrate the encoder–decoder RNN and reinforcement learning paradigms for abstractive summarization, taking advantages of both [18, 32]. For example, Paulus et al. [32] combined the maximum-likelihood cross-entropy loss with rewards from policy gradient reinforcement learning to reduce exposure bias. Liu et al. [18] proposed an adversarial process for abstractive text summarization, in which the generator is built as an agent of reinforcement learning. Yang et al. [42] trained an abstractive review summarization model in an end-to-end manner by exploring the encoder–decoder framework and multi-factor attentions.

2.2 Opinion summarization

In parallel, opinion summarization of product reviews has attracted increasing attention in recent years [9, 15, 24, 41, 46]. Generally, the opinion summarization approaches can be divided into two categories: aspect-based opinion summarization [39, 46] and non-aspect-based opinion summarization [8, 19]. In practice, aspect-based summarization can be more useful since it can present opinion distribution of each aspect separately. In particular, aspect-based opinion summarization usually consists of three distinct steps: aspect identification or feature selection [33, 50, 54], sentiment prediction [12, 44], and summary generation [26, 46, 54]. Gerani et al. [9] proposed an abstractive summarization system to product reviews by applying a template-based NLG framework and taking advantage of the discourse structure of reviews. Yu et al. [46] proposed a phrase-based approach which leveraged phrase properties to choose a subset of optimal phrases for generating the final summary. However, the above studies rely heavily on the selection of features, which are time-consuming and labor-intensive.

Different from the previous work, this study focuses on generating sentiment/aspect-aware abstractive review summarization that may better fit users' needs by using encoder–decoder framework with sentiment/aspect attentions.

2.3 Multi-task learning

Multi-task learning algorithms optimize multiple learning tasks simultaneously, and exploit the commonalities and differences across tasks, improving the generalization performance of the tasks [2, 7, 38, 49]. For example, Pasunuru and Bansal [31] improved video captioning by sharing knowledge with two related directed-generation tasks: a temporally directed unsupervised video prediction task and a logically directed language entailment generation task. Luong et al. [20] integrated multi-task learning with the encoder–decoder model, which shared the parameters of the encoder and decoder across the tasks. Substantial improvements have been shown in machine translation. In [40], Venugopalan et al. exploited complicated linguistic information in the decoder of video captioning by making use of an extra neural language model. Pasunuru et al. [31] improved the video caption performance by jointly learning the video captioning and the language entailment generation (auxiliary task). Multi-task learning is also popular in many other domains of computer vision and multimedia. For example, [47] proposed a spectral-spatial classification strategy for hyperspectral image classification, which mainly took advantages of multi-task learning to jointly learn the sparse representations and the stepwise MRF. Markatopoulou et al. [23] appended a multi-task learning loss to the convolutional neural network so that the model could capture both the label relations and the content relations between the tasks. Yang et al. [45] proposed multi-task spectral clustering method to jointly train several tasks and consider the intertask clustering correlation and intratask learning correlation.

2.4 Cross-domain sequence generation

Most previous cross-domain work belongs to the feature-based transfer, requiring manual selection of the pivot or non-pivot features [51–53]. Recently, domain adaptation is widely adopted in sequence generation, such as question answering [27], personalized dialogue generation [43], image captioning [4]. In [27], an initialization-then-adaptation strategy is adopted to learn a personalized task-oriented dialogue model. Yang et al. [43] extended the conventional encoder–decoder model to generated personalized dialogue generation via dual learning based domain adaptation. In [4], the adversarial training strategy is proposed to generate cross-domain captions by fully utilizing the labeled and unlabeled data. A critic and captioner components were trained in an adversarial manner, where the captioner tried to generate high-quality sequences while the critic network tried to distinguish them.

To date, no work is toward the aspect/sentiment-aware abstractive review summarization in domain adaptation scenario.

3 Our methodology

3.1 Problem definition

We use X^s and X^t to denote the collection of reviews in source and target domains respectively. Each input review $x = \{w_1, w_2, ..., w_{N_x}\}$ has a corresponding reference summary $y^s = \{w_1, w_2, ..., w_{N_y}\}$ and a binary domain label y^d (e.g., Food or Electronics domains), where N_x and N_y denote the sizes of the input document and the reference summary, respectively. Given an input review x, the abstractive review summarization task tries to generate a summary $\hat{y}^s = \{\hat{y}^s_1, \hat{y}^s_2, ..., \hat{y}^s_{N_s}\}$, where N_s denotes the length of the generated summary. For the domain classification task, given an input review x, our objective is to predict the domain label $\hat{y}^d$ for the input review. The domain classification model and the abstractive review summarization model work on a shared document encoding layer.

3.2 Domain classification in ASTR

In this section, we present a simple LSTM-based classification model for domain classification.

3.2.1 LSTM encoder

We convert each document $x = \{w_1, \ldots, w_{N_x}\}$ into a sequence of hidden states $H = \{h_1, \ldots, h_{N_x}\}$ by a single layer LSTM network, where N_x is the length of document x. The update of the hidden states of LSTM at time step t is computed as

$$h_t = \mathrm{LSTM}(\mathrm{h_{t-1}}, \mathrm{w_t}) \tag{1}$$

The reader can refer to [11] for the details of LSTM.

Aspect and lexicon representation learning by wsLDA In order to capture the relevant and discriminative information from a text piece in response to a given aspect and its sentiment polarity, we incorporate domain-specific aspect and sentiment lexicon background knowledge into the representation of a document. To that end, we first come up with a novel probabilistic generative model (weakly supervised LDA—wsLDA) to learn the domain-specific aspect and sentiment lexicon representations. Then, we develop a soft-attention mechanism to combine aspect (sentiment lexicon) knowledge and document representations so that our model is able to attend the aspect information and the corresponding sentiment.

The original LDA model is an unsupervised generative model to learn hidden thematic of structures in a large corpus of documents. Because of its unsupervised nature, LDA [1] can be used as a generic tool to group words into categories. Inspired by [28], our wsLDA model extends the standard LDA model, employing seed words to guide the topic model construction. The topics extract and categorize aspect terms and sentiment words automatically in the corresponding aspect and sentiment lexicon categories. It is thus able to best meet our specific needs—producing aspect and sentiment lexicon representations.

Suppose each document has three classes of topics: two *sentiment* topic, *K aspect* topics, and *M other* topics which are not related to sentiments and aspects, such as stopwords. Each document can be viewed as a mixture of the three classes of topics. Each topic is associated with a multinomial distribution over words. To prevent conceptual confusion, we use superscripts "*senti.*", "*aspect*" and "*other*" to indicate the variables that are related to *sentiment* topics, *aspect* topics and *other* topics, respectively. In addition, we assume that the vocabulary consists of V distinct words indexed by $\{1, \ldots, V\}$. For each document, we have three topic distributions $\theta^{senti.}$, θ^{aspect} and θ^{other} which represent the probabilities of *sentiment* topic n, *aspect* topic k and *other* topic m, respectively.

We use $\phi^{senti.}_{n,w}$, $\phi^{aspect}_{k,w}$ and $\phi^{other}_{m,w}$ to represent the probabilities of word w under *sentiment* topic n, *aspect* topic k and *other* topic m, respectively. The generation of *other* topics is similar to the original LDA model. Our approach is a weakly supervised approach since only some seed words for sentiment and aspect topics are needed to launch the process of generation of sentiment and aspect topics. For each *sentiment* topic n, its word distribution $\phi_n^{senti.}$ is chosen from a Dirichlet distribution $\text{Dir}(\beta_n^{senti.})$, where $\beta_n^{senti.}$ is a V-dimensional hyper-parameter that needs to be defined by users. The V-dimensional vector $\beta_n^{senti.}$ is computed by

$$\beta^{senti.}_{n,w} = \gamma_0(1-\omega_w)+\gamma_1\omega_w, \text{forw} \in \{1,\ldots,\text{V}\} \tag{2}$$

where $\omega_w = 1$ if the word w is a seed word in *sentiment* topic n; otherwise, we set $\omega_w = 0$. The scalars γ_0 and γ_1 are hyper-parameters. Intuitively, the biased prior $\beta_n^{senti.}$ enforces a seed word from *sentiment* topic n more probably to be generated from the *sentiment* topic n. The words distribution of each *aspect* topic $\phi_n^{aspect} \sim Dir(\beta_n^{aspect})$ is similarly constructed. The *sentiment* and *aspect* seed words have no intersection. In practical applications, asking users to provide some seeds is easy as they usually have a good knowledge of what is important in their domains.

For each word w in document x, a topic indicator λ is chosen from a topic class distribution p. The topic of the word w is then generated by $z_w \sim \text{Mult}(\theta^{(\lambda)})$, and the word itself is generated by $w \in \{1,\ldots,V\} \sim \text{Mult}(\phi_{z_w}^{(\lambda)})$. We summarize how the wsLDA model generates a corpus as follows (Dir and Mult mean Dirichlet and Multinomial distributions, respectively):

1. For *sentiment* topic $n \in \{0, 1\}$:
   (a) Draw a multinomial word distribution over words: $\phi_n^{senti.} \sim \text{Dir}(\beta_n^{senti.})$.
2. For each *aspect* topic $k \in \{1,\ldots,K\}$:
   (a) Draw a multinomial word distribution over words: $\phi_k^{aspect} \sim \text{Dir}(\beta_k^{aspect})$.
3. For each *other* topic $m \in \{1,\ldots,M\}$:
   (a) Draw a multinomial distribution over words: $\phi_m^{other} \sim \text{Dir}(\beta^{other})$.
4. For each document x in the corpus
   (a) Draw multinomial topic distributions $\theta^{senti.} \sim \text{Dir}(\alpha^{senti.})$, $\theta^{aspect} \sim \text{Dir}(\alpha^{aspect})$ and $\theta^{other} \sim \text{Dir}(\alpha^{other})$.
   (b) For each word of the document in $\{1,\ldots,N_x\}$, where N_x is the length of document x:
      i. Choose a topic class indicator $\lambda \sim \text{Mult}(p)$
      ii. Choose a topic z_w from $\text{Mult}(\theta^{(\lambda)})$,
      iii. Choose a word w from $\text{Mult}(\phi_{z_w}^{(\lambda)})$.
      iv. Emit word w

As an alternative representation, the graphical model of this generative process is shown in Fig. 1.

Given the hyper-parameters $\alpha = \{\alpha^{senti.}, \alpha^{aspect}, \alpha^{other}\}$, $\beta = \{\beta^{senti.}, \beta^{aspect}, \beta^{other}\}$ and p, our goal is to estimate the latent variables in the wsLDA model. Latent variables are unobservable, so they can only be inferred from other observed variables instead of directly measured. Here, the latent variables include λ, z, $\theta = \{\theta^{senti.}, \theta^{aspect}, \theta^{other}\}$, $\phi = \{\phi^{senti.}, \phi^{aspect}, \phi^{other}\}$. In this work, we employ the Gibbs sampling algorithm [34] to estimate the unknown parameters. We introduce some auxiliary notations. Let $\eta_{n,w}^{senti.}$ (or $\eta_{k,w}^{aspect}$, $\eta_{m,w}^{other}$) indicates the number of occurrences of *sentiment* topic n (or *aspect* topic k, *other* topic m) with word w in the whole corpus. Let $\delta_n^{senti.}$ (or δ_k^{aspect}, δ_m^{other}) indicates the number of occurrence of *sentiment* topic n (or *aspect* topic k, *other* topic m) in the current document. All these counts are defined excluding the current word w. Then, by the property of Dirichlet distribution [34], we can easily compute the latent variables. The reader may refer to [34] for a detailed derivation of the sampling procedure. In this paper, we aim to obtain the word distributions of *sentiment* and *aspect* topics. Mathematically, we compute $\phi^{senti.}$ and ϕ^{aspect} as:

$$\phi^{senti.}_{n,w} = \frac{\beta^{senti.}_{n,w} + \eta^{senti.}_{n,w}}{\sum_{w'=1}^{V}\left(\beta^{senti.}_{n,w'} + \eta^{senti.}_{n,w'}\right)}; \tag{3}$$

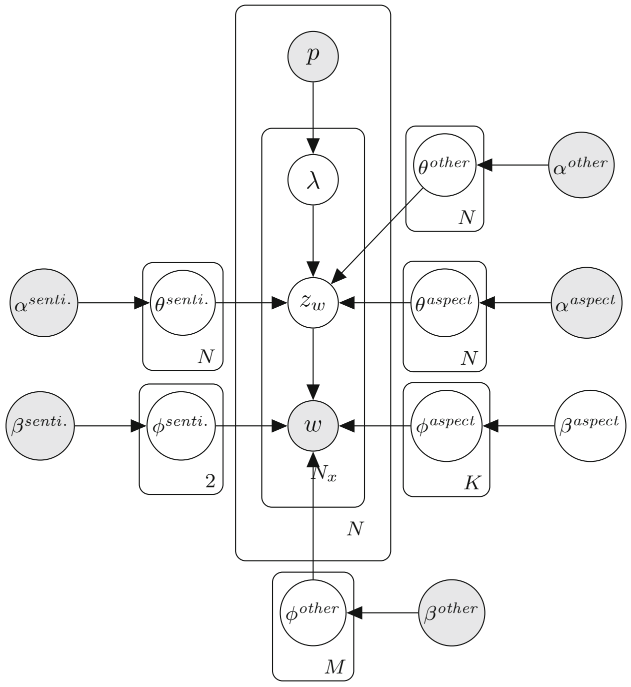

Fig. 1 The graphical model of weakly supervised LDA (wsLDA) model

$$\phi_{k,w}^{aspect} = \frac{\beta_{k,w}^{aspect} + \eta_{k,w}^{aspect}}{\sum_{w'=1}^{V} \left(\beta_{k,w'}^{aspect} + \eta_{k,w'}^{aspect} \right)} \tag{4}$$

In the Gibbs sampling procedure, we only need to maintain the counters $\eta^{senti.}$, η^{aspect}, η^{other}, $\delta^{senti.}$, δ^{aspect}, and δ^{other}, which takes $O(1)$ time to update for each iteration.

We further transform the V-dimensional word distributions of *sentiment* topic i and *aspect* topic j to *sentiment* embedding $u^{senti.}$ and *aspect* embedding u^{aspect} which are low-dimensional and have the same dimensions with the LSTM hidden states, by using a fully connected layer:

$$u_i^{senti.} = g(\phi_i^{senti.} \mathbf{W}^{senti.}) \tag{5}$$

$$u_j^{aspect} = g(\phi_j^{aspect} \mathbf{W}^{aspect}) \tag{6}$$

where the matrices $\mathbf{W}^{senti.} \in \mathbb{R}^{V \times dim}$ and $\mathbf{W}^{aspect} \in \mathbb{R}^{V \times dim}$ are trainable parameters, *dim* is the size of the hidden states of our LSTM encoder, g is the nonlinear function (i.e., tanh).

3.2.2 Aspect/sentiment-aware review representation

We design a multi-view attention mechanism to combine aspect (sentiment) knowledge and document representations so that our model is able to attend different parts of the document, responding to different aspects. In addition, our multi-view attention mechanism models overall semantics of the sentiment, aspect and context words, which helps to capture the important information from different representation subspaces at different positions.

Multi-view attention produces 2-dimensional attention weight matrix. Formally, with the representations of sentiment lexicon and aspect, the attention matrix $A \in \mathbb{R}^{b \times L}$ for the input document is computed as:

$$A = [A_1, A_2, \ldots, A_{N_x}] \tag{7}$$

$$A_i = \frac{\exp(\rho([h_i; u^{senti.}; u^{aspect}]))}{\sum_{j=1}^{L} \exp(\rho([h_j; u^{senti.}; u^{aspect}]))} \tag{8}$$

where N_x is the length of the input sequence, $A_i \in \mathbb{R}^b$ denotes the ith row of attention matrix which indicates the importance of the ith word in multiple hops of attention, b is the number of hops of attention, each row of attention matrix A denotes one hop of attention on the whole document (a single-view attention), and ρ is the attention function that calculates the importance of h_i in multiple hops of attention:

$$\rho([h_i; u^{senti.}; u^{aspect}]) = U_a^T \tanh(\mathrm{W_a}[\mathrm{h_i}; \mathrm{u^{senti.}}; \mathrm{u^{aspect}}]) \tag{9}$$

where U_a and W_a are projection parameters to be learned.

After computing the multi-head attention matrix for the input document, we can calculate the final aspect/sentiment-aware review representation emb_x for review x based on the multi-view attention matrix A as:

$$emb_x = flatten(AH_x) \tag{10}$$

where *flatten* is an operation that flattens matrix into the vector representation.

3.2.3 Domain prediction

The final aspect/sentiment-aware review representation emb_x is fed into a task-specific fully connected layer and a softmax layer for domain classification of the given review x:

$$\hat{y}^d = softmax(V_2 \cdot F_x) \tag{11}$$

$$F_x = \tanh(V_1 \cdot emb_x) \tag{12}$$

where V_1 and V_2 are projection parameters to be learned. The classifier is trained by minimizing the cross-entropy between the predicted distribution and the ground truth distribution:

$$L^{\text{domain}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{D} y_i^d \log(\hat{y}_i^d) \tag{13}$$

where D, number of domains; N, number of reviews in the training set.

3.3 Abstractive review summarization

The abstractive review summarization subtask shares the same review representation module with the domain classification subtask.

3.4 LSTM decoder

Inspired by [37], the pointer-generator network is adopted as the decoder to generate summaries, which is essentially a language model for estimating the contextual probability of the next word except that it is conditioned on the input. The pointer-generator network allows both copying words from input text via pointing (P_{vocab}), and generating words from a fixed vocabulary (P_{gen}). Thus, the pointer-generator has the ability to produce out-of-vocabulary (OOV) words.

On each step t, the decoder receives the word embedding of the previous word w_{t-1} (while training, this is the previous word of the reference summary; at test time it is the previous word emitted by the decoder) and update its hidden state s_t as:

$$s_t = \mathrm{LSTM}(\mathrm{s_{t-1}}, \mathrm{c_t}, \mathrm{w_{t-1}}) \tag{14}$$

The attention mechanism is used to calculate the attention weights a_t and context vector c_t as in [37], which computed as a weighted sum of the hidden states of the input text.

$$c_t = \sum_{n=1}^{N} \mu_{t,n} h_n \tag{15}$$

$$\mu_t = \mathrm{softmax}(\mathrm{e_t}) \tag{16}$$

$$e_{(t,i)} = \tanh(\mathrm{W_h emb_i + W_s s_t + b_{attn}}) \tag{17}$$

where W_h, W_s and b_{attn} are learnable parameters. The context vector c_t is then concatenated with the decoder state s_t and fed through a linear layer and a *softmax* layer to produce the vocabulary distribution $P_{vocab}(w_t)$, which provides us with our final distribution from which to predict word w_t.

In the pointer-generator model, the generation probability $p_{gen} \in [0, 1]$ for time step t is calculated from the context vector c_t, the decoder state s_t, and the decoder input x_t at time step t.

$$p_{gen} = \mathrm{sigmoid}(\mathrm{U_c^T c_t + U_s^T s_t + U_x^T x_t + b_{gen}}) \tag{18}$$

where vectors U_c, U_s, U_x and scalar b_{gen} are learnable parameters.

For each review, we let the extended vocabulary denote the union of the original vocabulary and all words appearing in the source review. We obtain the following probability distribution over the extended vocabulary:

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_t^i \tag{19}$$

Once we have defined the summarization model, we can estimate the parameters to minimize the negative log-likelihood of the training data by using mini-batch stochastic gradient descent:

$$L^{\mathrm{sum}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} \log P(w_t) \tag{20}$$

where N is the number of reviews in training set, T is the length of the sequence.

3.5 Joint training of ASTR model

Overall, our ASTR model consists of two subtasks, each has a training objective. For the purpose of strengthening the learning of the shared aspect/sentiment review representations, we train these two related tasks simultaneously.

$$L_{\mathrm{ML}} = \gamma_2 L^{\mathrm{domain}} + \gamma_3 L^{\mathrm{sum}} \tag{21}$$

where γ_2 and γ_3 are hyper-parameters that determine the weights of L_{domain} and L_{sum}. Here, we set $\gamma_2 = 0.2$, $\gamma_3 = 0.8$, which are determined by performing the grid search on a validation set.

3.5.1 Policy gradient for summary generation

However, the maximum-likelihood estimation (MLE) method suffers from two main issues. First, the evaluation metric is different from the training loss. Second, the input of the decoder at each time step is often the previous ground truth word during training. This exposure bias [35] leads to error accumulation at the testing phase. To alleviate the aforementioned issues when generating summaries, we also optimize directly for Rouge-1 since it achieves best results among the alternatives such as METEOR [14] and BLEU [30], by using policy gradient algorithm, and minimize the negative expected rewards:

$$L_{\mathrm{RL}}^{\mathrm{sum}} = (r(\hat{y}) - r(\bar{y})) \sum_{t}^{N_s} \log p(\bar{y}_t | y_{1:t-1}^s; x) \tag{22}$$

where $r(\hat{y})$ is the reward of a greedy decoding generated sequence $\hat{y}$, and $r(\bar{y})$ is the reward of sequence $\bar{y}$ generated by sampling among the vocabulary at each step.

After pre-training the proposed model by minimizing the joint ML objective (see Eq. 21), we switch the model to further minimize a mixed training objective, integrating the reinforcement learning objective J_{RL}^{sum} with the original multi-task loss J_{ML}:

$$L_{\mathrm{mixed}}(\Theta) = \mu L_{\mathrm{ML}} + (1 - \mu) L_{\mathrm{RL}}^{\mathrm{sum}} \tag{23}$$

where μ is a hyper-parameter, and we set $\mu = 0.1$. Θ denotes the set of parameters of the proposed model.

4 Experimental setup

4.1 Datasets description

We evaluate our model on Amazon reviews dataset from Stanford Network Analysis Project (SNAP) [25]. The raw dataset consists of 34,686,770 reviews from Amazon users spanning different kinds of products such as books, video games, food, music. To test the performance of our model in the cross-domain scenario, we use the 568,454 reviews from Food category as the source domain data and randomly choose 200,000 reviews from Electronics category as target domain data. Each review mainly contains product, user information, ratings, a plaintext review and a review summary. We randomly select 5000 reviews from each dataset as test data and validation data, respectively, and use the remaining for training.

4.2 Baseline methods

In the experiments, we compare our model with several strong baseline methods:

- *ABS* Attentional encoder–decoder recurrent neural networks for abstractive text summarization proposed in [29].
- *CopyNet* This model proposes a copying mechanism that is integrated into the neural sequence-to-sequence architecture [10].
- *LenEmb* This model uses the neural sequence-to-sequence model as the backbone and controls the output summary length [13].
- *PGC* The pointer-generator coverage networks proposed in [37] which copies words from the source text via pointing, while retaining the ability to produce novel words through the generator.
- *TraPGC* This model pre-trains the sequence-to-sequence framework of PGC model [37] on both the source and target data with an unsupervised learning setting.
- *DeepRL* The deep reinforced model (ML + RL version) proposed in [32],[1] which introduce a new objective function by combining the maximum-likelihood cross-entropy loss with rewards from policy gradient reinforcement learning to reduce exposure bias.
- *TraDeepRL* This model pre-trains the sequence-to-sequence model of DeepRL [32] on both the source and target data with an unsupervised learning setting.
- *GANsum* The generative adversarial network for abstractive summarization [18].
- *CGU* This a global encoding framework [17], which consists of a convolutional gated unit to perform global encoding to improve the representations of the source-side information.
- *HSSC* This is a hierarchical end-to-end model for jointly improving text summarization and sentiment classification [22].

4.3 Implementation details

In all experiments, data preprocessing is performed. Following the same strategy as in [29], we apply a minimal preprocessing step using PTB tokenization, lower-casing, and replacing of words seen less than 3 times with *UNK*. We limit the vocabulary to 20,000 most frequent words appearing in the training set. The source reviews and the summaries share the same vocabularies. We tune the hyper-parameters based on the performance on the validation sets.

We first settle down the implementation details for our weakly supervised LDA model. The numbers of *aspect* topics and *other* topics are set to $K = 5$ and $M = 10$, respectively. Hyper-parameter β^{other} is set to the typical value ($\beta^{other} = 0.1$), as suggested in previous work [28]. For hyper-parameters $\alpha^{senti.}$, α^{aspect}, α^{other}, γ_0 and γ_1, we vary their values from 0 to 1 with an increment of 0.05. Finally, we get $\alpha^{senti.} = \alpha^{aspect} = 0.25$, $\alpha^{other} = 0.1$, $\gamma_0 = 0.15$, γ_1 = 0.85.

For the parameters of the deep neural networks, we use 100-dimensional word2vec vectors pre-trained on English Wikipedia corpus to initialize the words in both datasets. Other parameters are randomly sampled from the uniform distribution $U(-0.01, 0.01)$. The sizes of the aspect/lexicon attention vectors and the LSTM encoder/decoder hidden layers are set to 300. *ASTR* model is trained using Adadelta with mini-batch. Other hyper-parameters include: learning rate 0.01, L2 regularization 0.001, dropout 0.2, batch size 64.

5 Experimental results

In this section, we compare our model with baseline methods from quantitative and qualitative perspectives.

5.1 Quantitative evaluation

Following the same evaluation as in [29], we compare our model with baseline methods in terms of Rouge-1, Rouge-2, Rouge-L, and Human evaluation.

Rouge-1 and Rouge-2 [16] are widely used evaluation metrics for summarization tasks, which estimate the consistency between *n*-gram occurrences in the generated and the reference summaries. Rouge-L compares the longest common sequence between the generated summary and the reference summary. For human evaluation, we evaluate the informativeness and fluency of the generated summaries by randomly select 200 examples from the test set. Similar to [5], three human evaluators were invited to score each summary generated by all models based on their informativeness (if the summary captures important information in the article) and fluency (if the summary is written in well-formed English). For the informativeness part, the human evaluator will read through the summaries to try to get a sense of what the story is talking about. If the human evaluator does not get an idea of what the original story is talking about, then this summary will be assigned a lower score. For the fluency part, if the summary is not well-written (e.g., grammatical mistakes), it will also have a lower score. Human evaluators are required to score the summaries by taking the above 2 factors into consideration, where 1 indicates the lowest score and 10 indicates the highest score. We report the model comparison in both out-of-domain setup and in-domain setups.

[1] We select their RL + ML model which obtains second highest ROUGE score but produces summaries of highest readability.

Out-of-domain abstractive review summarization To test the performance of our model in the cross-domain scenario, we train the proposed model on Food reviews (or Electronics reviews) and test it on Electronics reviews (or Food reviews). The experimental results of our ASTR model and the baseline methods are summarized in Tables 1 and 2. It clearly demonstrates that ASTR achieves better performance than the strong competitors. For example, our model improves 7.35% on Rouge-1, 10.82% on Rouge-2, 6.89% on Rouge-L, 11.47% on human evaluation when training the model on Food reviews and testing it on Electronics reviews.

In-domain abstractive review summarization We further report the experimental results in the in-domain setup like literature [29]. Concretely, we randomly choose 1000 reviews from Food reviews (or Electronics reviews) as test data, and the remaining reviews in Food review data (or Electronics review data) are used for training. We report the in-domain experimental results in Tables 3 and 4 for Food and Electronics review datasets, respectively. We observe that the proposed ASTR method substantially outperforms other methods and gets the state-of-the-art results on all evaluation metrics.

5.2 Ablation study

To investigate the effect of each component of the ASTR model, we also perform the ablation test of ASTR in terms of discarding domain categorization (i.e., w/o domain), aspect attention (i.e., aspect), and sentiment attention (i.e., sentiment). Due to the limited space, we only report the ablation text on Food data for both the out-of-domain and the in-domain setups. The results are summarized in Tables 5 and 6 for out-of-domain and in-domain setups, respectively. From the results, we can observe that generally all three factors contribute, and aspect attention contribute most. This is within our expectation since the aspect knowledge helps locate the salient information for abstractive summarization. The domain classification and sentiment attention also make the great contribution to abstractive review summarization, verifying that the domain and sentiment information plays a vital role in review summarization.

Table 1 Out-of-domain experiments (on food data)

Method	Rouge-1	Rouge-2	Rouge-L	Human evaluation
ABS	72.47	54.61	73.28	3.59
PGC	75.68	56.83	76.15	5.03
TraPGC	75.83	57.15	76.42	5.04
CopyNet	74.69	56.32	75.61	4.83
LenEmb	74.37	55.99	75.83	4.37
DeepRL	76.85	59.23	78.29	4.75
TraDeepRL	76.81	59.52	78.49	4.61
GANsum	77.13	59.07	79.06	5.32
CGU	75.58	56.74	76.19	4.49
HSSC	76.12	58.35	77.63	4.97
ASTR	79.56	62.26	81.93	5.93

Table 2 Out-of-domain experiments (on electronics data)

Method	Rouge-1	Rouge-2	Rouge-L	Human evaluation
ABS	54.32	35.83	50.95	3.42
PGC	57.64	37.82	53.15	5.14
TraPGC	57.93	38.05	53.42	5.23
CopyNet	57.27	37.65	53.22	4.98
LenEmb	56.95	37.14	52.43	4.43
DeepRL	58.42	37.93	54.26	4.62
TraDeepRL	58.56	37.83	54.47	4.83
GANsum	59.33	38.65	56.06	5.54
CGU	56.72	37.53	52.89	4.56
HSSC	57.83	37.49	53.25	4.97
ASTR	63.69	42.83	59.92	5.86

5.3 Qualitative evaluation

To evaluate the proposed model qualitatively, we reported some generated summaries by different models. ASTR model is trained on Food reviews data and test it on Electronics reviews data. Due to the limitation of space, we randomly choose two generated summaries by DeepRL and our model from test data for comparison. The results are reported in Table 7. We observe that ASTR tends to

Table 3 In-domain experiments (on food data)

Method	Rouge-1	Rouge-2	Rouge-L	Human evaluation
ABS	78.53	60.92	79.21	3.57
PGC	80.44	62.23	82.64	5.53
TraPGC	81.32	62.85	83.17	5.65
CopyNet	80.27	61.98	82.43	4.98
LenEmb	81.28	62.15	82.19	4.66
DeepRL	82.12	63.09	84.31	4.61
TraDeepRL	82.54	63.35	84.37	4.93
GANsum	82.48	63.65	84.13	6.12
CGU	81.53	62.67	82.95	4.85
HSSC	80.93	62.49	81.42	5.14
ASTR	83.92	65.06	86.43	6.24

Table 4 In-domain experiments (on electronics data)

Method	Rouge-1	Rouge-2	Rouge-L	Human evaluation
ABS	61.43	39.54	58.97	3.66
PGC	64.25	43.37	62.55	5.69
TraPGC	64.34	43.04	62.65	5.83
CopyNet	62.23	42.65	61.75	5.28
LenEmb	62.97	41.56	61.35	4.76
DeepRL	66.05	45.53	63.28	5.07
TraDeepRL	66.53	45.23	63.56	5.15
GANsum	66.78	44.95	63.46	5.87
CGU	63.95	41.86	61.17	4.86
HSSC	63.81	42.34	60.95	5.15
ASTR	68.45	46.76	65.92	6.13

generate more specific and meaningful summaries than the compared methods (especially the DeepRL method) in response to the given texts. For example, our model successfully catches the characteristics of the “sound” aspect of the headphones. The advantage of our model comes from its capability of integrating sentiment and aspect information into the attention encoder–decoder model.

5.4 Computational cost

We train our model on a single Tesla P100 GPU. ASTR takes about 10.5 h per epoch for both datasets. As revealed in [29], most compared baseline models take about 10 h per epoch on an average except the hierarchical attention model, which takes 12 h per epoch. All models typically converge within 15 epochs using the early stopping criterion based on the validation cost. The training time until convergence therefore varies between 6 and 8 days depending on the model. Generating summaries at test time is reasonably fast with a throughput of about 15 summaries per second on a single GPU, using a batch size of 1.

6 Conclusion

This work deals with a new problem in abstractive text summarization field aspect/sentiment-aware abstractive review summarization in domain adaptation scenario. We introduce a multi-task learning approach ASTR, which leverages the benefit of supervised deep neural networks as well as unsupervised probabilistic generative models to strengthen the representation learning. A weakly supervised LDA (wsLDA) model is proposed to automatically learn the domain-specific aspect and sentiment lexicon representations for both the source domain and target domain data, which are then fed into neural hidden states of the target words to form aspect-aware and lexicon-aware attentive review representations. The domain information and the attentive review representations are composed to perform the domain classification and abstractive review summarization. ASTR is evaluated on a real-life Amazon reviews dataset in out-of-domain as well as in-domain setups. The experiments demonstrate that NAACL has robust superiority over competitors and set state-of- the-art.

In future work, we will explore the dependency parser to further capture the long-range information for certain aspect and sentiment words which are relevant in syntax but far in word order. We also plan to boost the performance of abstractive review summarization by simulating the human reading cognitive process that consists of three stages: pre-reading, active reading, and post reading. If one desires to create a machine intelligence imitating such a reading comprehension skill of humans, studying these three-stage human reading cognitive process is quite necessary.

Table 5 Ablation test results for out-of-domain setup (on food data)

Method	Rouge-1	Rouge-2	Rouge-L	Human evaluation
ASTR	79.56	62.26	81.93	5.93
ASTR w/o domain	77.53	59.11	79.06	5.39
ASTR w/o aspect	78.48	59.75	80.20	5.63
ASTR w/o sentiment	78.13	60.34	80.56	5.71

Table 6 Ablation test results for in-domain setup (on food data)

Method	Rouge-1	Rouge-2	Rouge-L	Human evaluation
ASTR	83.92	65.06	86.43	6.24
ASTR w/o domain	81.73	63.25	83.47	5.84
ASTR w/o aspect	82.48	64.42	84.54	6.02
ASTR w/o sentiment	82.31	63.93	85.12	5.98

Table 7 Example summaries

Input “The drive I received was Sabrent model SBT UFDB fitted with a Mitsui drive. The drive installed completely plug and play and operated without any further actions to install it. The operation is a bit querky on some disks, all of which must be High Density (HD) 1.44 mb disks. I had one problem and discovered the write protection slide tab was open. Sabrents latest manual SFT USDB NewManual pdf, from their website, explains that the write protection tab must not be open (so you can see through it—for those of you that have not been through the glorious 35 floppy era) for the drive to function correctly. Also, I tried several techniques to get some floppies to work correctly—right clicked on folder and selected Open, ejected and replugged the drive without the floppy in it, and several verbal invectives. Most of my photos, program data files and old program installation files seemed to copy easily, albeit slowly, onto the hard drives. Overall, a good bargain compared to the hassle of trying to install a 3.5 floppy into the computer that wasn’t designed to accept such an installation. Also, the floppy is an acceptable media for transferring smaller files from computer to computer”

Ground truth “Program data files and old program installation files seemed to copy easily, albeit slowly”

DeepRL “The drive some disks must not be open work correctly hard install not open to install”

GANsum “The drive installed completely plug for the drive to work correctly”

ASTR “The drive function correctly program seemed copy easily albeit slowly good bargain”

Input “Purchased this to replace my 10 Sony ear buds. Sony is not bad, but sounds a little dry and boring. I was surprised to hear the sound improved even when connected to a PC. Bass sounds more natural (not over emphasized as woofers do), and mid-range sounds fuller. Yo-Yo Ma sounds better with this compare to Sony.Even though this is supposed to be on the comfortable side of headphones, I still feel uncomfortable after using it for a while. I think it is the case for almost any headphone. However, this is not the best for conversation. I prefer my cheap ear bud for Skype, which sounds a lot clearer. This is not a surprise since all telephones actually do filtration similar to that ear bud’s frequency response to make conversations clearer and save system resources. By the way, I got a chance to try a Koss TD-80. This Portapro beats TD hands down in sound quality. If you enjoy music, but not on an AM radio or very compressed files, give this a try. Update: to improve the clarity in conversation (Skype) or vocal, I cut a hole in the center of the foam pad. Now everything works beautifully. By the way, its cheaper to order replacement pads through Amazon. Koss charges 5 for a pair, while Amazon sells a pack for the same price”

Ground truth “Excellent for the price, realistic reproduction of sound”

DeepRL “Sony bad sounds better make clear amazon enjoy music make sounds clearer cheap ear”

GANsum “Bass sounds natural sounds fuller better for any headphone enjoy music works beautifully”

ASTR “Telephones make conversations clearer and save system resources, cheaper to replace for the same price”

Acknowledgements The work was partially supported by CAS Pioneer Hundred Talents Program, National Natural Science Foundation of China (No. 61750110516), and Guangdong Natural Science Fund Project (No. 2018A030313017).

References

1. Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022
2. Caruana R (1998) Multitask learning. In: Learning to learn. Springer, pp 95–133
3. Chen Q, Zhu X, Ling Z, Wei S, Jiang H (2016) Distraction-based neural networks for modeling documents. In: Proceedings of the international joint conference on artificial intelligence
4. Chen T-H, Liao Y-H, Chuang C-Y, Hsu W-T, Fu J, Sun M (2017) Show, adapt and tell: adversarial training of cross-domain image captioner. IEEE Int Conf Comput Vis (ICCV) 2:521–530
5. Cheng J, Lapata M (2016) Neural summarization by extracting sentences and words. In: Proceedings of the 54th annual meeting of the association for computational linguistics, Berlin, Germany. Association for Computational Linguistics, vol 1: long papers, pp 484–494
6. Chopra S, Auli M, Rush AM (2016) Abstractive sentence summarization with attentive recurrent neural networks. In: The 15th annual conference of the north American chapter of the association for computational linguistics: human language technologies, pp 93–98
7. Collobert R, Weston J (2008) A unified architecture for natural language processing: deep neural networks with multitask learning. In: The international conference on machine learning, pp 160–167
8. Ganesan K, Zhai CX, Han J (2010) Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In: The 23rd international conference on computational linguistics. ACL, pp 340–348
9. Gerani S, Mehdad Y, Carenini G, Ng RT, Nejat B (2014) Abstractive summarization of product reviews using discourse structure. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1602–1613
10. Gu J, Lu Z, Li H, Li VOK (2016) Incorporating copying mechanism in sequence-to-sequence learning. ACL 1:1631–1640
11. Hochreiter S, Schmidhuber J (1997) Long short-term memory. In: Neural computation, pp 1735–1780
12. Hu M, Liu B (2006) Opinion extraction and summarization on the web. AAAI 7:1621–1624
13. Kikuchi Y, Neubig G, Sasano R, Takamura H, Okumura M (2016) Controlling output length in neural encoder-decoders. In: EMNLP, pp 1328–1338
14. Lavie A, Agarwal A (2007) Meteor: an automatic metric for MT evaluation with high levels of correlation with human judgments. In: Proceedings of the second workshop on statistical machine translation. Association for Computational Linguistics, pp 228–231
15. Li F, Han C, Huang M, Zhu X, Xia Y-J, Zhang S, Yu H (2010) Structure-aware review mining and summarization. In: Proceedings of the 23rd international conference on computational linguistics. Association for Computational Linguistics, pp 653–661
16. Lin CY (2004) Rouge: a package for automatic evaluation of summaries. In: ACL workshop on text summarization branches out, vol 8

17. Lin J, Sun X, Ma S, Su Q (2018) Global encoding for abstractive summarization. In: IJCAI
18. Liu L, Lu Y, Yang M, Qu Q, Zhu J, Li H (2018) Generative adversarial network for abstractive text summarization. In: Association for the advancement of artificial intelligence
19. Lu Y, Zhai C (2008) Opinion integration through semi-supervised topic modeling. In: Proceedings of the 17th international conference on World Wide Web. ACM, pp 121–130
20. Luong MT, Le QV, Sutskever I, Vinyals O, Kaiser L (2016) Multi-task sequence to sequence learning. In: International conference on learning representations
21. Ly DK, Sugiyama K, Lin Z, Kan MY (2011) Product review summarization from a deeper perspective. In: Annual international ACM/IEEE joint conference on digital libraries. ACM, pp 311–314
22. Ma S, Sun X, Lin J, Ren X (2018) A hierarchical end-to-end model for jointly improving text summarization and sentiment classification. In: IJCAI
23. Markatopoulou F, Mezaris V, Patras I (2016) Deep multi-task learning with label correlation constraint for video concept detection. In: Proceedings of the 2016 ACM on multimedia conference. ACM, pp 501–505
24. Mason R, Gaska B, Durme BV, Choudhury P, Hart T, Dolan B, Toutanova K, Mitchell M (2016) Microsummarization of online reviews: an experimental study. In: AAAI, pp 3015–3021
25. McAuley J, Leskovec J (2013) Hidden factors and hidden topics: understanding rating dimensions with review text. In: ACM conference on recommender systems. ACM, pp 165–172
26. Mei Q, Ling X, Wondra M, Su H, Zhai CX (2007) Topic sentiment mixture: modeling facets and opinions in weblogs. In: Proceedings of the 16th international conference on World Wide Web. ACM, pp 171–180
27. Mo K, Li S, Zhang Y, Li J, Yang Q (2017) Personalizing a dialogue system with transfer learning. In: The thirty-first AAAI conference on artificial intelligence
28. Mukherjee A, Liu B (2012) Aspect extraction through semi-supervised modeling. In: ACL, pp 339–348
29. Nallapati R, Zhou B, Gulcehre C, Xiang B et al (2016) Abstractive text summarization using sequence-to-sequence RNNS and beyond. In: Proceedings of the 20th SIGNLL conference on computational natural language learning. Association for Computational Linguistics, pp 280–290
30. Papineni K, Roukos S, Ward T, Zhu WJ (2002) BLEU: a method for automatic evaluation of machine translation. In: The 40th annual meeting on association for computational linguistics, pp 311–318
31. Pasunuru R, Bansal M (2017) Multi-task video captioning with video and entailment generation. In: The 55th annual meeting of the association for computational linguistics, pp 1273–1283
32. Paulus R, Xiong C, Socher R (2017) A deep reinforced model for abstractive summarization. arXiv preprint arXiv:1705.04304
33. Popescu AM, Etzioni O (2007) Extracting product features and opinions from reviews. In: Natural language processing and text mining. Springer, pp 9–28
34. Porteous I, Newman D, Ihler A, Asuncion A, Smyth P, Welling M (2008) Fast collapsed gibbs sampling for latent Dirichlet allocation. In: SIGKDD, pp 569–577
35. Ranzato MA, Chopra S, Auli M, Zaremba W (2015) Sequence level training with recurrent neural networks. arXiv preprint arXiv:1511.06732
36. Rush AM, Chopra S, Weston J (2015) A neural attention model for abstractive sentence summarization. In: Proceedings of the 2015 conference on empirical methods in natural language processing. Association for Computational Linguistics, pp 379–389
37. See A, Liu PJ, Manning CD (2017) Get to the point: summarization with pointer-generator networks. In: Proceedings of the 55th annual meeting of the association for computational linguistics. Association for Computational Linguistics, pp 1073–1083
38. Shen W, Zhao K, Jiang Y, Wang Y, Bai X, Yuille A (2017) Deepskeleton: learning multi-task scale-associated deep side outputs for object skeleton extraction in natural images. IEEE Trans Image Process 26(11):5298–5311
39. Titov I, McDonald R (2008) Modeling online reviews with multi-grain topic models. In: Proceedings of the 17th international conference on World Wide Web. ACM, pp 111–120
40. Venugopalan S, Hendricks LA, Mooney R, Saenko K (2016) Improving LSTM-based video description with linguistic knowledge mined from text. In: Proceedings of the conference on empirical methods in natural language processing, pp 1961–1966
41. Yang M, Mei J, Xu F, Tu W, Lu Z (2016) Discovering author interest evolution in topic modeling. In: Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 801–804
42. Yang M, Qu Q, Shen Y, Liu Q, Zhao W, Zhu J (2018) Aspect and sentiment aware abstractive review summarization. In: Proceedings of the 27th international conference on computational linguistics, pp 1110–1120
43. Yang M, Zhao Z, Zhao W, Chen X, Zhu J, Zhou L, Cao Z (2017) Personalized response generation via domain adaptation. In: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 1021–1024
44. Yang M, Zhu D, Rashed M, Chow KP (2014) Learning domain-specific sentiment lexicon with supervised sentiment-aware LDA. In: Proceedings of the twenty-first European conference on artificial intelligence (ECAI'14), pp 927–932
45. Yang Y, Ma Z, Yang Y, Nie F, Tao Shen H (2015) Multitask spectral clustering by exploring intertask correlation. IEEE Trans Cybern 45(5):1083–1094
46. Yu N, Huang M, Shi Y et al (2016) Product review summarization by exploiting phrase properties. In: Proceedings of the 26th international conference on computational linguistics (COLING), pp 1113–1124
47. Yuan Y, Lin J, Wang Q (2016) Hyperspectral image classification via multitask joint sparse representation and stepwise MRF optimization. IEEE Trans Cybern 46(12):2966–2977
48. Zhang L, Zhang Y, Chen Y (2012) Summarizing highly structured documents for effective search interaction. In: SIGIR. ACM
49. Zhang Q, Levine MD (2016) Robust multi-focus image fusion using multi-task sparse representation and spatial context. IEEE Trans Image Process 25(5):2045–2058
50. Zheng W, Zhu X, Wen G, Zhu Y, Yu H, Ganv J (2018) Unsupervised feature selection by self-paced learning regularization. Pattern Recognit Lett. https://doi.org/10.1016/j.patrec.2018.06.029
51. Zheng W, Zhu X, Zhu Y, Hu R, Lei C (2017) Dynamic graph learning for spectral feature selection. Multimed Tools Appl 77(22):29739–29755
52. Zhu X, Zhang S, Hu R, Zhu Y et al (2018) Local and global structure preservation for robust unsupervised spectral feature selection. IEEE Trans Knowl Data Eng 30(3):517–529
53. Zhu X, Zhang S, Li Y, Zhang J, Yang L, Fang Y (2018) Low-rank sparse subspace for spectral clustering. IEEE Trans Knowl Data Eng. https://ieeexplore.ieee.org/document/8417928
54. Zhuang L, Jing F, Zhu XY (2006) Movie review mining and summarization. In: Proceedings of the 15th ACM international conference on information and knowledge management. ACM, pp 43–50

Affiliations

Min Yang[1] · Qiang Qu[1] · Ying Shen[2] · Kai Lei[3] · Jia Zhu[4]

[1] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, People's Republic of China

[2] School of Electronics and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen, People's Republic of China

[3] Shenzhen Key Lab for Information Centric Networking and Blockchain Technology, School of Electronics and Computer Engineering, Peking University, 518055 Shenzhen, People's Republic of China

[4] School of Computing Science, South China Normal University, Guangzhou, People's Republic of China