World Wide Web (2019) 22:1401–1425
https://doi.org/10.1007/s11280-018-0591-0

A weakly-supervised extractive framework for sentiment-preserving document summarization

Yun Ma[1] · **Qing Li**[1,2]

Received: 15 December 2017 / Revised: 11 April 2018 / Accepted: 17 May 2018 /
Published online: 30 May 2018

Abstract The popularity of social media sites provides new ways for people to share their experiences and convey their opinions, leading to an explosive growth of user-generated content. Text data, owing to the amazing expressiveness of natural language, is of great value for people to explore various kinds of knowledge. However, considerable user-generated text contents are longer than what a reader expects, making automatic document summarization a necessity to facilitate knowledge digestion. In this paper, we focus on the reviews-like sentiment-oriented textual data. We propose the concept of Sentiment-preserving Document Summarization (SDS), aiming at summarizing a long textual document to a shorter version while preserving its main sentiments and not sacrificing readability. To tackle this problem, using deep neural network-based models, we devise an end-to-end weakly-supervised extractive framework, consisting of a hierarchical document encoder, a sentence extractor, a sentiment classifier, and a discriminator to distinguish the extracted summaries from the natural short reviews. The framework is weakly-supervised in that no ground-truth summaries are used for training, while the sentiment labels are available to supervise the generated summary to preserve the sentiments of the original document. In particular, the sentence extractor is trained to generate summaries i) making the sentiment classifier predict the same sentiment category as the original longer documents, and ii) fooling the discriminator into recognizing them as human-written short reviews. Experimental results on two public datasets validate the effectiveness of our framework.

Keywords document summarization · sentiment-preserving · end-to-end neural network · reinforcement learning

✉ Yun Ma
yunma3-c@my.cityu.edu.hk

Qing Li
itqli@cityu.edu.hk

[1] Department of Computer Science, City University of Hong Kong, Hong Kong, SAR, China

[2] Multimedia software Engineering Research Centre, City University of Hong Kong, Hong Kong, SAR, China

1 Introduction

In recent years, the booming social media services have made users become the main characters on the Internet. Every day, huge amounts of data are contributed by users on social media sites, providing rich data resources for exploration. In particular, review data is of great value since it conveys the sentiments of users for objects such as products, movies, and restaurants. For example, the merchants can predict the market prospect and propose sale strategies by understanding the needs of customers; users may decide to whether or not watch a movie by referring to others' comments. However, the textual content of the reviews can be much longer than one may expect, making automatic document summarization an imperative tool to help grasp the main idea and save time.

Most existing text summarization techniques are applied to formal text data like news [12, 38], that is of relatively consistent style and more factual with neutral sentiment. However, it is quite different and much tougher to summarize review data, considering that reviews are generated by ordinary users and characterized by the implied sentiments [30]. On one hand, from the perspective of review readers, a good summarization of a review document should preserve the original sentiment, so as to keep the most important function as a review. On the other hand, the diverse writing styles and vocabularies of review data complicate the process of semantic extraction, making it difficult for learning algorithms to effectively recognize the patterns and generate a summary. Moreover, even for the formal text, automatic summarization is still a challenging task despite the progress made recently by deep learning algorithms, due to the gap between the information that natural language possesses and what the models are able to learn [12, 27, 33, 38].

Our work is related with the topic of sentiment summarization, or opinion summarization, which has been extensively studied in recent years. Sentiment summarization aims to creating a summary for a group of documents commenting on a specific object (e.g., a product) [17]. Existing works in this field mainly address the problem with aspect-based approaches [1, 13, 21, 43], where the summary is organized by various aspects and a list of positive and negative descriptions are presented for each aspect. Our work, in contrast, considers the summarization from a different perspective. Instead of conducting a vertical summarization for a document set by aspects, we target at single-document summarization with the goal of providing an abstract for each long document which is concise enough and preserves the main sentiments.

In this paper, we propose the concept of Sentiment-preserving Document Summarization (SDS) for sentiment-orientated text data like reviews. Specifically, given a long sentiment-orientated document, SDS aims at summarizing the document as a shorter version with preserved sentiments. Intuitively, to achieve this goal, we need to devise a model with two components, i.e., a summarizer to generate the shortened text, and a sentiment analyzer to justify the sentiment consistency. Regardless of the promising significant benefits in practice, it can be quite a non-trivial task to solve the problem in a supervised setting since it is expensive to obtain the ground-truth sentiment-preserving summary. In this paper, to address such an issue, we propose to deal with SDS in a weakly-supervised and extractive setting. By "weakly-supervised", we mean that we do not need the ground-truth summary, and only exploit the easy-to-obtain sentiment labels of the original documents since most reviews come with a rating indicating their overall sentiment for the commented object. By "extractive", we mean

that the summaries are created by selecting sentences from the original documents, instead of generating new text units (e.g. words and phrases) as in the abstractive setting which is hardly feasible without the ground-truth summaries. However, only with the sentiment-consistent constraint, we may achieve summaries unnaturally patched up with sentences, resulting in low-level readability. To tackle this problem, inspired by the works [23, 25, 47] introducing Generative Adversarial Networks (GAN) [11] in Natural Language Processing (NLP) tasks, we employ a discriminator to differentiate the generated short summaries and the natural short reviews written by human. Adversarially training the summarizer and the discriminator can eventually force the summarizer to generate summaries indistinguishable with human-written short reviews. Therefore, the objective of the summarizer is two-fold: 1) to generate summaries preserving the sentiments of the documents; and 2) to generate summaries natural as human-written short reviews.

Based on the above motivations, we develop an end-to-end neural network based framework to conduct weakly-supervised extractive sentiment-preserving document summarization. Our framework is composed of four parts: a hierarchical document encoder, a sentence extractor, a sentiment classifier, and a discriminator. Specifically, the hierarchical document encoder utilizes the multi-level structure of documents, from word-level to sentence-level and document-level, so as to better capture the semantics; the sentence extractor learns to select salient sentences to form the summary; the sentiment classifier predicts the sentiment of a given document and acts as a global assessment for the extractive summary; and the discriminator distinguishes the generated summaries from the human-written short documents. In particular, the sentence extractor is optimized to generate summaries which make the sentiment classifier predict consistent sentiment class with the original document and make the discriminator recognize them as natural human-written short documents. Unfortunately, since the sentence extracting procedure is discrete, a main challenge of the framework lies in the difficulty to backpropagate the gradients from the global sentiment supervision (given by the sentiment classifier) and the readability supervision (given by the discriminator) to the sentence extractor. We tackle this challenge by resorting to the policy gradient based reinforcement learning technique [40], where the sentence extractor is an agent to learn a policy for selecting sentences by maximizing the reward given by the sentiment classifier and the discriminator. The main contributions of this paper are listed below:

- We propose the concept of Sentiment-preserving Document Summarization (SDS) for digesting the long sentiment-orientated textual data. To the best of our knowledge, this is the first piece of work on single-document summarization for the reviews-like data.
- We propose an end-to-end weakly-supervised extractive framework to extract the salient sentences from documents by preserving their sentiments and not sacrificing readability. Reinforcement learning is utilized to tackle the discreteness of the sentence extracting process.
- We demonstrate the effectiveness of our framework on two public datasets, i.e., a movie review dataset from IMDB and a business review dataset from Yelp.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents the concept of SDS. In Section 4, we introduce the end-to-end neural network-based framework with its individual components. Section 5 reports the experimental results of the

proposed framework on the IMDB dataset and the Yelp dataset, followed by a conclusion and future work in Section 6.

2 Related work

The proposed sentiment-preserving document summarization is closely related to four research topics, i.e. text summarization, sentiment classification, sentiment summarization, and GAN applications in NLP tasks. In this section, we review existing works in terms of these four directions.

2.1 Text summarization

The task of text summarization can be categorized into extractive summarization and abstractive summarization. Extractive summarization identifies the salient text units from the original text and then concatenates them as the summary. Abstractive summarization, which is a much more tough task, first comprehends the text and then generates the summary in a paraphrasing manner, which means that it is not a necessity for text units in summary to occur in the original document.

Most of the traditional works focus on the extractive summarization, which is treated as a sentence selecting task. Exploiting useful hand-crafted features, such as the position and length of sentence, word overlap with the title, word frequencies in sentence, part-of-speech (POS) of words in sentence, sentence similarities, etc. [10, 34, 36], a saliency score is assigned to each sentence. Kupiec et al. [20] scored each the sentence independently using a binary classifier. In [6], Conroy et al. addressed problem as a sequence labeling task using the hidden Markov model. Mihalcea et al. [28] developed a TextRank algorithm to estimate the significance of sentences with a graph-based model. TextRank can be seen as an variant of the Page-Rank algorithm, where the weights of edges are normalized sentence similarities. Woodsend et al. [44] extracted the salient phrases and optimize the rearrangement by constrained integer linear programming (ILP).

During the past few years, since deep learning models have been successfully applied to many natural language processing tasks, researchers started to pay more attention to the abstractive summarization task. Typically, the abstractive summarization is tackled by the sequence-to-sequence framework consisting of an encoder and a decoder, where the encoder embeds the document as one or a set of latent vectors, and then the decoder emits the summary tokens one after another based on the encoded semantics and its current states. Inspired by the neural machine translation work [2] by Bahdanau et al., Rush et al. [38] developed the first data-driven model for abstractive summarization with an attention-based encoder-decoder network. Nallapati et al. [32] further introduced the large vocabulary trick [14], rich word features like TF-IDF, switching generator-pointer, and hierarchical attentions to address some learning difficulties. Gu et al. proposed the CopyNet [12] with a copy mechanism during the decoding process. Specifically, for each step, the model generates a token by summing up the probability of generating from the vocabulary and copying from the original text. See et al. [39] further extended the copy mechanism as a soft combination of generation and copy, where the weights of the two terms are predicted by the model.

The deep neural network based approaches are not restricted to abstractive summarization, and have also shown significant improvements on extractive summarization. Cheng et al. [5]

employed a hierarchical document encoder and a sentence extractor which sequentially generate labels for each sentence indicating the probability of the sentence selected to be in the summary. Nallapati et al. [33] proposed an extractive model by representing the information content, salience, novelty, and position importance using the hidden vectors, making the model much more interpretable.

To achieve better performance for text summarization based on the neural sequence-to-sequence framework, some works introduced more components in addition to the basic encoder and decoder. Miao et al. [27] incorporated a reconstructor to reconstruct the original text from the generated summary, so as to make the summary as informative as possible. Paulus et al. [35] proposed to combine the maximum likelihood loss with a global evaluation metric like ROUGE to guide the decoding. Cao et al. [4] linked the task of text classification and the extractive summarization with a shared document embedding. In our work, we also jointly learn the classification model and the summarization model, while we differentiate by using classification as an global supervision for the summarization rather than simply sharing internal representations.

In this paper, we focus on the extractive summarization for the proposed sentiment-preserving document summarization. The main reason is, under a weakly-supervised setting where the ground-truth summary is absent, extractive summarization is more realistic considering the readability of the generated summary.

2.2 Sentiment classification

The problem of sentiment classification has been dealt with by lots of works for a long time. Similar to text summarization, sentiment classification re-attracts the attention of researchers with the popularity of neural network-based deep learning algorithms.

Existing neural network-based models for sentiment classification typically generate a document embedding using a Convolutional Neural Network (CNN)-based or Recurrent Neural Network (RNN)-based network, on top of which is a Multi-Layer Perceptron (MLP) as the classifier. Kim [16] dealt with sentence classification by applying multiple convolutional filters of different sizes to extract the features from the text, and obtained the sentence embedding by a global pooling. Kalchbrenner et al. [15] used multiple convolutional layers to obtain different levels of abstraction and introduced the K-Max pooling strategy. Zhang et al. [48] further explored the finer-granularity character-level convolutional networks trained on large corpus for the classification task. Dieng et al. [9] combined the RNN-based model with a topic model, in which the RNN aimed to capture the local semantics while the topic model aimed to capture the global semantics. Compared with the flat structure in the above models, some works employed the tree structure of sentences obtained by constituency parsing or dependency parsing so as to capture the long-distance relationship. Tai et al. [41] proposed a tree-structured Long Short Term Memory (LSTM) to extract the sentence embedding. Mou et al. [31] developed the tree-based convolution for each node in the parsing tree by treating the parent node and children nodes in a different way.

Since documents are composed by smaller units like words and sentences, some works model a document in a hierarchical way, which typically includes a sentence-level encoder and document-level encoder. Tang et al. [42] utilized the LSTM or CNN to process the sentence level and the gated recurrent neural network for the document level. The word/sentence hidden states are averaged to get the sentence/document states. In contrast, Yang et al. [46] introduced the attention mechanism by giving different weights for different words/sentences, then the

sentence/document embeddings are computed as the weighted average of their lower-level elements.

Considering that the labelled data is not always available, strategies like pre-training, semi-supervised learning, and multi-task learning have been proposed. Dai et al. [7] proposed to pre-train a language model or an auto-encoder with unlabeled data, and similarly, Kiros et al. [19] devised a Skip-Thought model for pretraining which encodes a sentence first and then predicts its surrounding sentences. Xu et al. [45] introduced a variational auto-encoder as an auxiliary task to better model the semantics in the document. Liu et al. [24] developed a multi-task learning algorithms for more than ten sentiment classification tasks and incorporate adversarial training to extract the shared patterns among different tasks.

In this paper, similar to [42, 46], we encode the documents using a hierarchical RNN-based network; however, we share the encoder for the sentiment classifier and the sentence extractor to better correlate the two objectives as well as to reduce parameters.

2.3 Sentiment summarization

The topic of sentiment summarization, or opinion summarization, has been widely investigated in recent years with the booming of social media sites. Unlike the proposed sentiment-preserving document summarization which is a single-document summarization task, the sentiment summarization is generally considered as a multi-document summarization problem [17], that is, generating a summary for a set of review documents commenting on a specific object.

Most existing works focus on the aspect-based sentiment summarization, which creates a summary of positive and negative opinions for each aspect of the commented object (e.g., the picture quality of a camera). Hu et al. [13] addressed the aspect-based sentiment summarization with a three-step pipeline: aspect identification, mention extraction and sentiment prediction for each aspect, and summary generation. Following such a pipeline, Titov et al. [43] proposed a Latent Dirichlet Allocation (LDA) based model, called Multi-Aspect Sentiment model, by jointly modelling text and sentiment ratings. Amplayo et al. [1] also adopted a LDA-based model for aspect mention extraction together with a three-level sentiment classifier to generate a summarized sentiment score for each aspect. Lerman et al. [21] investigated the evaluation of various sentiment summarization models, based on which they further developed a ranking SVM based summarizer.

Beineke et al. [3] explored sentiment summarization for single documents by identifying a set of features for locating text segments representing the sentiments of the long documents. However, they aimed at extracting only one sentence expressing the opinion, instead of a set of sentences forming a reasonable summary of the document as in our case.

2.4 GAN applications in NLP tasks

The Generative Adversarial Network proposed by Goodfellow et al. [11] have achieved excellent performance in the computer vision field owing to its capability to generate realistic images. The classical GAN model is formulated as a min-max game with two players, i.e., a discriminator and a generator. The discriminator tries to distinguish real data samples from the fake ones created by the generator, while the generator tries to generate samples fooling the discriminator.

Recently, several works have applied the idea of generative adversarial training into NLP tasks. Li el al. [23] proposed an adversarial training framework for open-domain dialogue generation, in which a discriminator is introduced to differentiate the human generated message-response pairs with the model generated ones. Yang et al. [47] adopted a similar strategy for the machine translation task. Closely related with our work, Liu et al. [25] improved the performance of abstractive summarization for news by incorporating a discriminator to distinguish the ground-truth summary with the generated ones.

3 Sentiment-preserving document summarization (SDS): problem definition

In this section, we propose the concept of sentiment-preserving document summarization, aiming at summarizing a long sentiment-orientated document without loss of its sentiments. As mentioned in Section 1, a good sentiment-preserving summary is supposed to be: 1) consistent with the original long document in sentiments; and 2) indistinguishable from the short reviews written by human.

Formally, let d denote a long sentiment-orientated document which consists of a sequence of M sentences $\{s_i \mid i \in \{1, 2, \ldots, M\}\}$, and each sentence s_i consists of a sequence of N_i words $\{w_{ij} \mid j \in \{1, 2, \ldots, N_i\}\}$. The sentiment of d, c_d, is not given but falls into one of the possible K classes, and p_{short} denotes the data distribution of human-written short reviews. The objective of Sentiment-preserving Document Summarization (SDS) is to generate a summary $\hat{d}$ of d, which is composed of $\hat{M}$ sentences $\{\hat{s}_i | i \in \{1, 2, \ldots, \hat{M}\}\}$, with $\hat{M} < M$, and each $\hat{s}_i$ is composed of $\hat{N}_i$ words $\{\hat{w}_{ij} | j \in \{1, 2, \ldots, \hat{N}_i\}\}$, subject to 1) *the sentiment-consistency constraint:* that the sentiment of $\hat{d}$ is consistent with that of d, and 2) *the readability constraint:* that the probability density of $\hat{d}$ in the generated summary distribution p_g is equal to that in p_{short}, i.e., $p_g\left(\hat{d}\right) = p_{short}\left(\hat{d}\right)$.

Since the sentiment labels are relatively easy to obtain, we assume they are always available during model learning (but not in the inference phase). Depending on the availability of the ground-truth sentiment-preserving summaries, the set-up of the SDS problem can be categorized into strongly-supervised setting and weakly-supervised setting. Under the strongly-supervised setting with both the ground-truth $\hat{d}$ and ground-truth c_d, a general objective is to maximize the conditional probability $P\left(\hat{d}|d\right)$ with the sentiment-consistency constraint $c_{\hat{d}} = c_d$ and the readability constraint $p_g\left(\hat{d}\right) = p_{short}\left(\hat{d}\right)$. In this situation, $\hat{d}$ serves as a local supervision to guide the generation of sentences and words, which makes both extractive and abstractive solutions feasible, and the sentiment-consistency constraint and the readability constraint serve as global supervisions to justify that the generated summary preserves the sentiment and has good readability. Actually, in this strongly-supervised setting, the benefits of the readability can be incremental comparing with the supervision from the ground-truth summary. However, since the ground-truth sentiment-preserving summaries are difficult to obtain, more often a weakly-supervised setting is preferred where only the sentiment labels are available. In this situation, the readability constraint become more significant for preventing the system from generating unnatural summaries by simply linking the sentiment indicators. Furthermore, abstractive summarization appears to be inviable in this weakly-supervised setting as there is no guarantee that the generated text can act as a summary for the original document; instead, an extractive framework is more suitable as to be described in Section 4.

4 Weakly-supervised extractive framework for SDS

In this section, we present the proposed weakly-supervised extractive framework for SDS. The overview of this end-to-end model is introduced first, followed by an elaboration of its individual components and the learning techniques.

4.1 Overview

Under the weakly-supervised and extractive setting for SDS, assume that we have two datasets: D_l, containing the long sentiment-orientated documents sampled from the long documents distribution $p_{long}(d)$ for summarization, and D_s containing the human-written short sentiment-orientated documents sampled from the short documents distribution $p_{short}(d)$ for training the discriminator. Given a long sentiment-orientated document $d \in D_l$, the extractive framework generates a sentiment-preserving summary $\hat{d}$, where each sentence in the summary comes from the sentences in the original document. Specifically, the model aims to learn four components, i.e., a) a hierarchical document encoder $E = \{E^S, E^D\}$ where E^S is the sentence-level encoder to learn a representation for each sentence based on their word-level representations, while E^D is the document encoder to learn a representation for the document based on the sentence-level representations from E^S; b) a sentence extractor G which estimates a probability to be selected to form the sentiment-preserving summary and then generates the summary by filtering a certain number of sentences; c) a sentiment classifier C which discriminates the sentiment class based on the document representation; and d) a discriminator F which estimates the probability that a given document is a human-written short review instead of a summary generated from a long document.

The flow of the proposed framework is shown in Figure 1. A short document $d \in D_s$ goes through two paths, i.e., $d \rightarrow E^S \rightarrow E^D \rightarrow C$ and $d \rightarrow F$. The first path hierarchically encodes the document from word-level representations to sentence-level representations (by E^S) and a document-level representation (by E^D), on top of which the sentiment classifier C makes its prediction on sentiment categories, while the second path classifies the document as a human-written short document or a generated summary from a long document. A long document $d \in D_l$ goes through three paths, i.e., $d \rightarrow E^S \rightarrow E^D \rightarrow C$, $d \rightarrow E^S \rightarrow G \rightarrow E^D \rightarrow C$, and $d \rightarrow E^S \rightarrow G \rightarrow F$. The first path, same as the first path for short documents, outputs the prediction on sentiment categories for d. The second path and the third path share the beginning sub-path $d \rightarrow E^S \rightarrow G$, where the sentence-level representations from E^S are fed to G, thus each sentence is assigned a selecting probability and a summary $\hat{d}$ for d is formed. With the generated summary $\hat{d}$, the second path further encodes it into the document-level representation with E^D and predicts its sentiment category using C, while the third path classifies it as a human-written short document or a generated summary from a long document with F.

In this framework, based on the five paths (i.e., two paths for the short documents and three paths for the long documents) in Figure 1, we involve five primary loss functions, i.e., a cross entropy loss on sentiment prediction for short documents $J_{ce-short}$, a cross entropy loss on sentiment prediction for long documents $J_{ce-long}$, a cross entropy loss on sentiment prediction for summarized long documents $J_{ce-summary}$, a binary cross entropy loss from the "human-written short documents vs. generated summaries from long documents" discrimination for short documents $J_{bce-short}$, and a binary cross entropy loss from the "human-written short documents vs. generated summaries from long

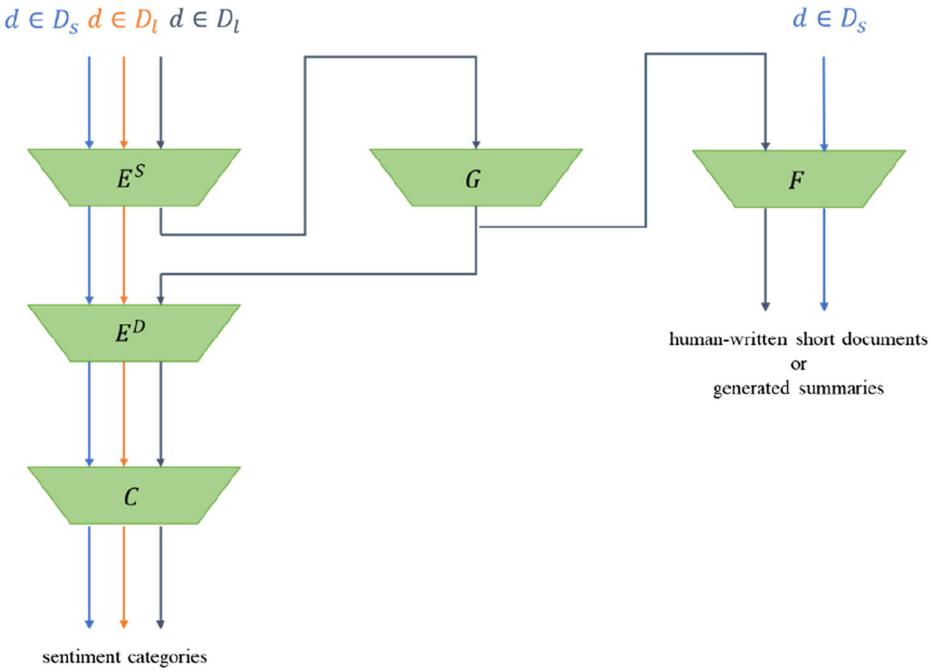

Figure 1 Architecture overview of the end-to-end extractive framework for SDS under a weakly-supervised setting

documents" discrimination for summarized long documents $J_{bce-summary}$. The formulations for these five losses are listed as follows:

$$J_{ce-short} = E_{d\sim p_{short}(d)}\left[-\log P_C\left(E^D\left(E^S(d)\right)\right)_{c_d}\right] \tag{1}$$

$$J_{ce-long} = E_{d\sim p_{long}(d)}\left[-\log P_C\left(E^D\left(E^S(d)\right)\right)_{c_d}\right] \tag{2}$$

$$J_{ce-summary} = E_{d\sim p_{long}(d)}\left[-\log P_C\left(E^D\left(G\left(E^S(d)\right)\right)\right)_{c_d}\right] \tag{3}$$

$$J_{bce-short} = E_{d\sim p_{short}(d)}\left[-\log P_F(d)\right] \tag{4}$$

$$J_{bce-summary} = E_{d\sim p_{long}(d)}\left[-\log\left(1-P_F\left(G\left(E^S(d)\right)\right)\right)\right] \tag{5}$$

where $P_C(\cdot)$ is the predicted probability distribution from C on all the potential sentiment classes, and $P_F(\cdot)$ is the probability of the input document being from the human-written short documents distribution $p_{short}(d)$. By limiting these losses to related components, we formulate the cost functions for each component, i.e., E, G, C and F.

For the sentence extractor G, we minimize

$$J_G = J_{ce-summary} - \lambda_G J_{bce-summary} \tag{6}$$

where the first term forces the extractor to generate summaries from which the correct sentiment can be inferred using C, the second term forces the extractor to generate natural summaries fooling the discriminator F, and λ_G is the hyper-parameter balancing the two terms.

For the sentiment classifier C, we minimize:

$$J_C = J_{ce-short} + J_{ce-long} + \lambda_C J_{ce-summary} \tag{7}$$

where the first term is to fit the (short document, sentiment) pairs $\{(d, c_d) | d \sim p_{short}(d)\}$, the second term is to fit the (long document, sentiment) pairs $\{(d, c_d) | d \sim p_{long}(d)\}$, and the third term, weighted by the hyper-parameter λ_C, is to fit the (summarized long document, sentiment) pairs $\left\{ \left(\hat{d}, c_d \right) | d \sim p_{long}(d) \right\}$. Since G tends to extract irrational summaries in the beginning, we initialize λ_C as a 0; as training proceeds, G is capable of generating reasonable sentiment-preserving summaries, so we increase λ_C to a positive value and train C by augmenting the training data with the (summarized long document, sentiment) pairs.

For the discriminator F, we minimize:

$$J_F = J_{bce-short} + J_{bce-summary} \tag{8}$$

where the first term tries to assign high probability to human-written short documents, and the second term tries to assign low probability to the generated summaries for long documents, so as to separate these two data distributions.

For the document encoder E, we minimize

$$J_E = J_{ce-short} + J_{ce-long} + \lambda_E J_{ce-summary} \tag{9}$$

where the three terms are the same with those in J_C, and λ_E is a hyper-parameter weighting $J_{ce-summary}$. While the sentence-level encoder E^S is also involved in the loss $J_{bce-summary}$, we omit this term from the objective function since we did not observe any benefits when including it in our preliminary experiments. As the training proceeds, we increase the value λ_E to a positive value from 0 as in the sentiment classifier case.

Considering that the process of summary generation involves discrete steps, learning the sentence extractor via direct gradient backpropagation becomes impossible. To address this problem, we resort to the reinforcement learning technique (to be discussed in Section 4.3.1). Besides, since both the supervision signals from the sentiment classifier and from the discriminator for the sentence extractor are global supervisions, we introduce an additional ranking loss for G to provide better local guidance by exploiting the gradients of C with respect to the sentence representations, as to be discussed in Section 4.3.2.

4.2 Model components

4.2.1 Document encoder

Consider that the documents are featured with a hierarchical structure, i.e. a document is composed of a sequence of sentences, while a sentence is composed of a sequence of words, we employ a hierarchical document encoder to capture such compositionality. Specifically, the hierarchical

document encoder consists of two sub-components, i.e., a sentence-level encoder E^S to encode words as sentences, and a document-level encoder E^D to encode sentences as the document.

While both CNN-based models and RNN-based models can be exploited for the encoding sub-components, i.e., E^S and E^D, our preliminary experiments show the RNN-based models perform better for our task. In particular, for both the sentence-level encoder E^S and the document-level encoder E^D, we adopt the bi-directional LSTM model, which is a variant of the vanilla RNN. Given the input x_t at the current timestamp t, and the cell state c_{t-1} and hidden state h_{t-1} at the previous timestamp $t-1$, the LSTM cell updates the cell state and hidden state for the current time stamp as follows:

$$f_t = \sigma\left(W_f[x_t, h_{t-1}] + b_f\right) \tag{10}$$

$$i_t = \sigma(W_i[x_t, h_{t-1}] + b_i) \tag{11}$$

$$o_t = \sigma(W_o[x_t, h_{t-1}] + b_o) \tag{12}$$

$$\tilde{c}_t = tanh(W_c[x_t, h_{t-1}] + b_c) \tag{13}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{14}$$

$$h_t = o_t \odot \tanh(c_t) \tag{15}$$

where f_t is the forget gate controlling how much information of the previous time stamp is kept for the current state, i_t is the input gate controlling how much information from the new input signal is accepted, and o_t is the output gate controlling how much information is exposed in h_t; $\sigma(\cdot)$ denotes the sigmoid function, $\tanh(\cdot)$ denotes the hyperbolic tangent function, and $\odot$ represents element-wise product. The weight matrices W_f, W_i, W_o, W_c and the bias vectors b_f, b_i, b_o, b_c are the parameters to be learned. The bi-directional LSTM consists of a forward LSTM which reads the input sequence from the first element (e.g. a word or a sentence) to the last one, and a backward GRU which reads from the last to the first. As a result, each element x_t in the sequence will get a forward hidden state $\overrightarrow{h_t}$ and a backward hidden state $\overleftarrow{h_t}$, which are further concatenated to form x_t's hidden presentation $h_t = \left[\overrightarrow{h_t}, \overleftarrow{h_t}\right]$. As a result, given a sentence s_i with its word-level representations $\{\boldsymbol{w_{ij}}\}$, the sentence-level encoder outputs a set of hidden representations $\left\{h_{ij}^W\right\}$ corresponding to each word, based on which we get the sentence-level representation:

$$\boldsymbol{s_i} = \frac{1}{N_i}\sum_{j=1}^{N_i} h_{ij}^W \tag{16}$$

Similarly, with the sentence-level representations $\{\boldsymbol{s_i}\}$, the document-level encoder outputs a set of hidden representations $\left\{h_i^S\right\}$ corresponding to each sentence, and the document-level representation is obtained by

$$\boldsymbol{d} = \frac{1}{M}\sum_{i=1}^{M} h_i^S \tag{17}$$

The architecture of the hierarchical document encoder is shown in Figure 2, which is able to better capture the semantics of the document, especially for the long-distance relationships.

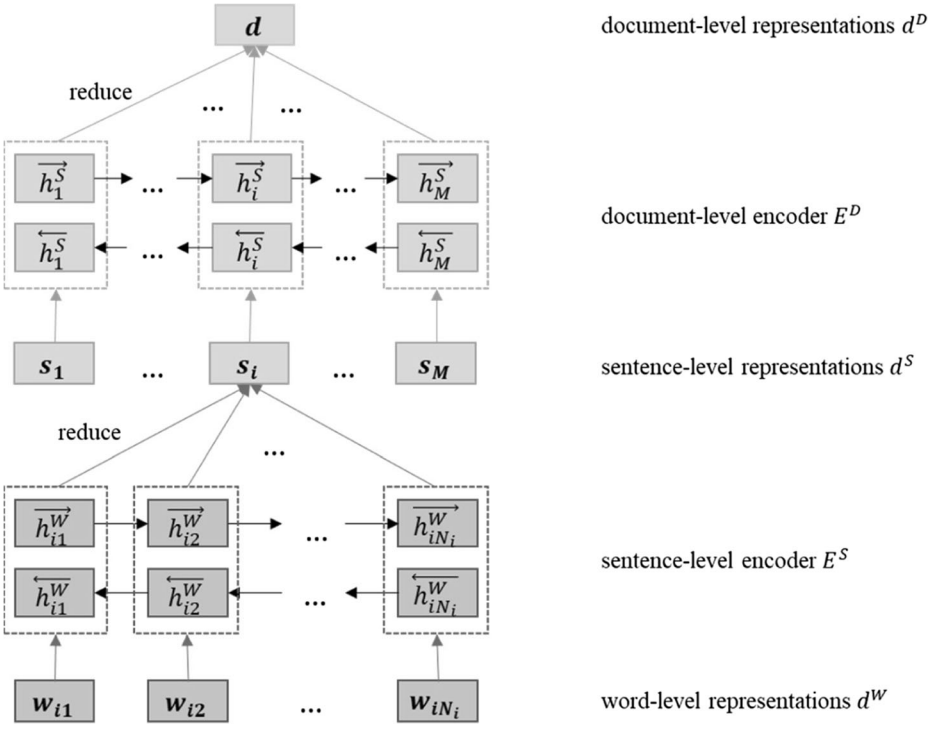

Figure 2 Architecture of the hierarchical document encoder

4.2.2 Sentence extractor

The sentence extractor G is to generate a sentiment-preserving summary by selecting $\hat{M}$ sentences, with $\hat{M} < M$, from a long document $d \in D_l$. As the first step, a selecting probability is estimated for each sentence s_i based on its sentence-level representation $\boldsymbol{s_i}$. We achieve this by using a MLP model:

$$P_G(z_{s_i} = 1|s_i) = \sigma(MLP(\boldsymbol{s_i})) \tag{18}$$

where $z_{s_i} \epsilon \{0, 1\}$ represents the selection label (1 means selected and 0 means not), and $\sigma(\cdot)$ is the sigmoid function. While it is possible to apply a more complex model, like the one in [5] which utilizes another LSTM model with a MLP, we found this simple approach works well in our scenario. With the estimated probabilities, the second step creates the summary by selecting sentences. We adopt different strategies for training and inference. During training, we treat the selection result for each sentence s_i as a random sample from the Bernoulli distribution parameterized by its selecting probability $P_G(z_{s_i} = 1|s_i)$, aiming to learn the model in a larger search space. During inference, however, the summary is formed by filtering sentences with higher probabilities as the most reliable result. Considering the various lengths of the original documents, we create the summary with a predefined percentage ρ. Given a document with M sentences, the sentence extractor selects the top $M\rho$ sentences with the highest selecting probabilities as the summary. As the last step, in both the sampling and top-$M\rho$ strategies, the selected sentences are concatenated as in their original order, instead of the order by the value of the selecting probability, to provide better consistency with the original

document. The created summary $\hat{d}$ will be encoded by the document-level encoder to generate the document-level representation $\hat{\boldsymbol{d}}$.

4.2.3 Sentiment classifier

The sentiment classifier C predicts a probability distribution over the K possible sentiment classes for a given document using its document-level representation. Similar to the sentence extractor, we employ the MLP model with a softmax output layer to achieve this:

$$P_C(\boldsymbol{d})_k = \frac{\exp(v_k)}{\sum_{k'=1}^{K}\exp(v_{k'})} \tag{19}$$

where

$$v = MLP(\boldsymbol{d}) \tag{20}$$

4.2.4 Discriminator

The discriminator F predicts the probability of a given document d being from the distribution of human-written short documents. We implement the discriminator with an architecture composed by a hierarchical RNN (which is in the same structure with the hierarchical document encoder in Section 4.2.1 but with independent parameters) and a MLP with a sigmoid output activation function. Specifically, the hierarchical RNN encodes the input document d as a latent representation d^F, and then the probability of d being a human-written short document is estimated as:

$$P_F(d) = \sigma\left(MLP\left(d^F\right)\right) \tag{21}$$

4.3 Learning techniques for sentence extractor

4.3.1 Policy learning

Since the discreate step involved in the sentence extractor, i.e. selecting sentences based on the estimated probabilities, disables the backpropagation of gradient from the supervision given by the sentiment classifier and the discriminator, a main challenge in this framework is how to optimize the parameters in G. We solve this difficulty through the reinforcement learning technique, where the sentence extractor is regarded as an agent trying to maximize a reward R provided by the sentiment classifier and the discriminator. Reinforcement learning has been widely employed in situations like non-differential functions or discrete actions, where regular techniques like gradient backpropagation will be stuck. Recently, [35] utilized the reinforcement learning technique to optimize the abstractive summarization model with respect to the ROUGE metric. [45] used reinforcement learning to optimize a sequence generator with the supervision signal from a global discriminator.

In our framework, we define the reward R for a generated summary $\hat{d}$ for a long document d as:

$$R\left(\hat{d}\right) = P_C\left(\hat{d}\right)_{c_d} + \lambda_{G\_reward} P_F\left(\hat{d}\right) \tag{22}$$

where the first term is the probability assigned to the correct sentiment category by the sentiment classifier, and the second term is the probability with which the discriminator recognize it as being from the human-written short document distribution. Furthermore, we adopt the self-critical policy gradient learning technique as in [35, 37]. Formally, given a document d and the selecting probabilities $\{P_G(z_{s_i}=1|s_i)\}$ estimated by the sentence extractor G, two summaries $\hat{d}$ and $\hat{d}_b$ are generated. $\hat{d}$ is generated by the Bernoulli sampling as described in Section 4.2.2, while $\hat{d}_b$ is a deterministic output with a threshold function. Specifically, for $\hat{d}_b$, a sentence s_i is selected if $P_G(z_{s_i}=1|s_i)$ is larger than 0.5. Let $z_{s_i}\left(\hat{d}\right)$ denote the actual selection label in the summary $\hat{d}$, $R\left(\hat{d}\right)$ denote the reward for the action "selecting $\hat{d}$ as the sentiment-preserving summary" of the sentence extractor agent, and $R\left(\hat{d}_b\right)$ denote the reward for the action "selecting $\hat{d}_b$ as the sentiment-preserving summary". The parameters in G can be optimized by minimizing

$$J_{G-RL}=\left(R\left(\hat{d}\right)-R\left(\hat{d}_b\right)\right)\sum_{i=1}^{M}-z_{s_i}\left(\hat{d}\right)logP_G(z_{s_i}=1|s_i)-\left(1-z_{s_i}\left(\hat{d}\right)\right)\log(1-P_G(z_{s_i}=1|s_i)) \tag{23}$$

In this formulation, $R\left(\hat{d}_b\right)$ is treated as a bias term to reduce variance and encourage summary results achieving higher reward than the thresholding deterministic selection.

4.3.2 Ranking loss

Currently, the supervision signals from both the sentiment classifier and the discriminator are too global for the extractor whose job is to assess the selection probability for each sentence. In other words, the loss is computed for the created summary which is a sequence of sentence selections, yet no explicit guidance is provided for estimated salience score of an individual sentence. Therefore, we introduce a ranking loss to give local supervisions for the sentence extractor.

The motivation here is to force the sentence extractor to assign high probabilities for salient sentences and low probabilities for the non-salient sentences. Unfortunately, under the weakly-supervised setting, no ground-truth telling the salience of sentences is available. Inspired by works in neural network visualization adopting the first-derivative to measure the contribution of the input units [22], we propose pseudo salience by using the gradient of the sentiment classifier with respect to the sentences. Formally, the pseudo salience score for a sentence s_i is in a long document d

$$q(S_i) \quad = \quad \|\nabla_{s_i} J_c\|_2 \tag{24}$$

where J_{ce − long}[d] is the cross entropy loss on sentiment prediction for the document d.

Sentences making a significant contribution to the classification decision tend to have a high salience score. Let L denote the subset of sentences with largest n_L salience scores. We categorize the sentences to salient sentences $S^+=\{s_i \mid s_i\in L\}$ and non-salient sentences $S^-=\{s_i \mid s_i\notin L\}$, and randomly sample U $\langle salient, non-salient\rangle$ pairs $\{\langle s_{u1}, s_{u2}\rangle | s_{u1}\in S^+, s_{u2}\in S^-\}$. The ranking loss is defined as

$$J_\gamma=\frac{1}{U}\sum_{u=1}^{U}\max(0,\xi-P_G(z_{s_{u1}}=1|s_{u1})+P_G(z_{s_{u2}}=1|s_{u2})) \tag{25}$$

where ξ is the margin. In our experiment, we dynamically set n_L as 30% of the total number of sentences in each document, $U=2$, and $\xi=0.5$. By minimizing the ranking loss, the sentence

extractor can obtain a local supervision for assessing the significance of individual sentences. As a result, the objective of G is reformulated to minimize

$$J_G = J_{G\text{–}RL} + \lambda_{G\_rank} J_\gamma \tag{26}$$

where λ_{G_rank} is the hyper-parameter to balance the reinforcement learning loss and the ranking loss.

5 Experiments

In this section, we apply the proposed weakly-supervised extractive framework for sentiment-preserving document summarization to real-world datasets. We first introduce the datasets used for training, then describe the training details, and report the results with discussions.

5.1 Dataset

Recall that in the weakly-supervised setting for SDS only the sentiment labels for the documents are available. We train our proposed framework on two publicly available datasets, i.e., a movie review dataset from IMDB [8] and a business review dataset from Yelp Dataset Challenge 2013 [42].

The IMDB dataset has 348,415 reviews with a vocabulary of 115,831 unique words, and each movie review is associated with a sentiment score ranging from 1 to 10, where 1 is the most negative and 10 is the most positive. The average number of sentences and words in a review are 14.0 and 325.6, respectively, while the maximum can reach 148 and 2802, respectively. We use the same 280,593/33,793/34,029 train/valid/test split as in [8]. Considering ten levels of sentiment labels are in a quite fine granularity, we also create a variant by mapping the sentiment labels in the original dataset to three sentiment levels: negative, neutral, and positive. Specifically, the labels 1, 2, 3, 4 are mapped to the negative class, the labels 5 and 6 are mapped to the neutral class, and the labels 7, 8, 9, 10 are mapped to the positive class. We discriminate the two versions by naming the original dataset as IMDB-10 and the variant with three sentiment levels as IMDB-3 in the following.

The Yelp dataset has 335,018 reviews with a vocabulary of 211,245 unique words, and each review is associated with a sentiment score ranging from 1 to 5, where 1 is the most negative and 5 is the most positive. The average number of sentences and words in a review are 8.9 and 151.6, respectively, while the maximum can reach 151 and 1184, respectively. We use the same 268,013/33,501/33,504 train/valid/test split as in [42].

5.2 Training details

Since we target at the long documents for extractive summarization and require a short documents set for the discriminator training, we split out datasets into two parts based on the document length. In particular, for each dataset, we categorize the documents with more than τ sentences as the long documents, and the documents with less than or equal to τ sentences as the short documents. We empirically set $\tau = 5$, considering documents with no more than 5 sentences are pretty easy for reading and make the summarization unnecessary. As a result, the IMDB dataset is divided into a long documents dataset D_l with a 247,411/29,800/29,937 train/valid/test split and a short documents dataset D_s with a 33,182/3993/4092 train/

valid/test split; the Yelp dataset is divided into a long documents dataset D_l with a 164,986/20,693/20,611 train/valid/test split and a short documents dataset with a 103,027/12,808/12,893 train/valid/test split.

For both the IMDB dataset and the Yelp dataset, sentence splitting and tokenization are conducted using Stanford CoreNLP [26]. Words occurring less than 5 times are replaced with an <UNK> token, resulting a vocabulary of size 97,223 for the IMDB dataset and 44,558 for the Yelp dataset, respectively. We initialize the word embeddings by learning a Skip-Gram model [29] using training data.

In our experiments, we tune our hyper-parameters using the validation set. The document encoder (of both the sentence-level and the document-level) utilizes the single-layer LSTMs for the forward LSTM and backward LSTM. The dimension of word representation is set to 200, while the hidden size for both the sentence-level encoder and the document-level encoder is set to 100, with 50 for the forward LSTM and 50 for the backward LSTM. The sentence extractor is a MLP with one hidden layer using ReLU as the non-linear activation, and the hidden size is 100. The sentiment classifier is a one-layer feed-forward network with the softmax output layer. The discriminator includes a hierarchical RNN in the same structure with the document encoder and a one-layer feed foward network with sigmoid output activation. We ramp-up both λ_C and λ_E from 0 to 0.5, set $\lambda_{G-reward}$=1, λ_{G-rank}=0.2. The model is optimized using Adam [18] algorithm with an initial learning rate of 0.001. The batch size is 128. Gradient clipping is employed and the maximum gradient norm is set to 1. The implementation uses the PyTorch deep learning framework and all experiments are conducted on a single GeForce GTX 1080 GPU.

5.3 Results and discussions

5.3.1 Summarization results

Table 1, Table 2, Table 3, and Table 4 show the sentiment-preserving extractive summarization results with our proposed framework on a movie review from the IMDB-3 dataset (Table 1), a movie review from the IMDB-10 dataset (Table 2), and two restaurant reviews from the Yelp dataset (Tables 3 and 4). For each exemplar review document, we provide the summaries with 20 and 50% extraction ratios.

First of all, we can observe that our weakly-supervised extractive summarization framework is able to select the salient sentences representing the user's core idea, regardless of the position of these sentences in the document. In particular, the three sentences forming the 20% summary for the movie review in Table 1, i.e., "it was that bad! ", "i 've seen some terrible horror movies in my time, but this was the absolute worst! ", and "the only bright point in this awful mess was wes ramsey." are from two leading positions and a near ending position; while the single sentence forming the 20% summary for the restaurant review in Table 4, i.e., "i would highly recommend the dried peppers from all over the world and in levels of hotness.", lies in a middle position. This is obviously a desirable property for user-generated data. Unlike the formal textual data like news which usually puts important content in the beginning, user-generated text like reviews may not have fixed styles, disabling the classical "selecting the leading N sentences" method.

Secondly, the sentiment classifier predicts close, if not the same, sentiment categories for the original review documents and the summarized versions, proving that the summaries indeed preserve the sentiments. Although the sentiment score of the 20% summary for the

Table 1 Sentiment-preserving extractive summarization results with 20 and 50% summarization ratios on the exemplar review document from the IMDB-3 dataset

Original review Document **Sentiment: Negative**	<s> it was that bad! <s> i 've seen some terrible horror movies in my time, but this was the absolute worst! <s> first of all, the plot did not make . <s> why would dracula want to impregnate a human woman to make of "super vampires? " <s> their offspring would obviously be half-human . <s> and the guy who played dracula was awful . <s> no offense to heavier people, but dracula was described as tall and thin . <s> this actor, andrew whatever his name was, just did not fit the part . <s> he looked like a washed-up wwf pro wrestler . <s> he struggled with his fangs through the movie and i thought that he was going to 1 (spit them out or 2) swallow them . <s> the actress who played elizabeth seemed like a valley girl who, you know, was shopping at the mall for some kewl clothes and uh, got lost at the mall and wandered into the home of umm, this really creepy, fat vampire dude . <s> and what was the point of the scene with `` bram stoker "(wes ramsey) and the french-speaking people who were trying to kill him? <s> they were talking about the guillotine, and the revolution had ended a century before! <s> they definitely needed to get out more . <s> the only bright point in this awful mess was wes ramsey . <s> he is handsome and personable and does have talent . <s> quick wes, cross this off your resume and find a new agent! **Predicted sentiment: Negative**
Sentiment-preserving summary (20%)	<s> it was that bad! <s> i 've seen some terrible horror movies in my time, but this was the absolute worst! <s> the only bright point in this awful mess was wes ramsey . **Predicted sentiment: Negative**
Sentiment-preserving summary (50%)	<s> it was that bad! <s> i 've seen some terrible horror movies in my time, but this was the absolute worst! <s> first of all, the plot did not make . <s> the actress who played elizabeth seemed like a valley girl who, you know, was shopping at the mall for some kewl clothes and uh, got lost at the mall and wandered into the home of umm, this really creepy, fat vampire dude. <s> they were talking about the guillotine, and the revolution had ended a century before! <s> they definitely needed to get out more . <s> the only bright point in this awful mess was wes ramsey . <s> quick wes, cross this off your resume and find a new agent! **Predicted sentiment: Negative**

movie review in Table 4 is classified as 4 instead of 5, it has extracted the most salient sentence representing the user's main opinion for the restaurant. Such summaries, with only 20 and 50% of the original contents, can enable readers to have a quick grasp for what the authors think. Moreover, this is useful for high-efficiency required applications like online sentiment analysis.

Thirdly, our proposed framework flexibly allow different percentage of compression during inference with the trained model. Comparing the 20% summaries and the 50% summaries, we can observe that the detailed information increases with the summarization ratio. For the movie review in Table 2, the 20% summary generally expresses the author's praise to the movie and the filmmaker, while the 50% summary tells more details about the author's feelings and how

Table 2 Sentiment-preserving extractive summarization results with 20 and 50% summarization ratios on the exemplar review document from the IMDB-10 dataset

Original review Document	<s> put simply, not only the greatest silent film ever made, but one of the 10-15 perfect films .
Sentiment: 10	<s> sunrise, to me, is the definitive moment in silent cinema .
	<s> not only is sound unnecessary, but so are words -- indeed, there are remarkably few title cards .
	<s> instead, murnau trusts in the ability of his images to convey his story; he does n't need words .
	<s> the story itself is simple, archetypal .
	<s> it functions primarily as a frame onto which murnau scene after scene of breathtaking splendor .
	<s> in particular, the first shots of the city are dizzyingly complex and layered .
	<s> additionally, it 's impossible to come away unimpressed by the storm which tosses the characters during their return journey .
	<s> murnau is one of the few filmmakers, and perhaps the first, to truly embrace the possibilities of film as its own medium, rather than as a novelty or, alternatively, a convenient way to preserve a stage play .
	<s> though he is better remembered for other films, most particularly nosferatu, sunrise is his crowning achievement .
	Predicted sentiment: 10
Sentiment-preserving summary (20%)	<s> put simply, not only the greatest silent film ever made, but one of the 10-15 perfect films .
	<s> murnau is one of the few filmmakers, and perhaps the first, to truly embrace the possibilities of film as its own medium, rather than as a novelty or, alternatively, a convenient way to preserve a stage play .
	Predicted sentiment: 10
Sentiment-preserving summary (50%)	<s> put simply, not only the greatest silent film ever made, but one of the 10-15 perfect films .
	<s> sunrise, to me, is the definitive moment in silent cinema .
	<s> in particular, the first shots of the city are dizzyingly complex and layered .
	<s> murnau is one of the few filmmakers, and perhaps the first, to truly embrace the possibilities of film as its own medium, rather than as a novelty or, alternatively, a convenient way to preserve a stage play .
	<s> though he is better remembered for other films, most particularly nosferatu, sunrise is his crowning achievement .
	Predicted sentiment: 10

he/she was attracted by the scenes. For the restaurant review in Table 3, the 20% summary describes the most intolerable thing of the restaurant, while the 50% summary includes more details on how the user dislike the restaurant.

Fourthly, the readability of extracted summaries varies with different summarization ratios. With 20% summarization ratio, the readability is acceptable for all the four examples, however, the summaries are not always natural as human-written short reviews. For example, the 20% summary in Table 2 is much less coherent than the 20% summary in Table 1. In contrast, the summaries under 50% summarization ratio are much more satisfying. For all the four illustrative examples, it is difficult to tell whether it is a model-generated extractive summary or written by human reviewers.

Ablation study Since the discriminator in our proposed framework is introduced to supervise the sentence extractor to generate natural summaries with good readability, we conduct an ablation study by training a model without using the discriminator. Table 5 presents a movie review from the IMDB dataset and the results with 50% summarization ratio achieved by the proposed model and the ablated variant where the discriminator is

Table 3 Sentiment-preserving extractive summarization results with 20 and 50% summarization ratios on the exemplar review document from the Yelp dataset

Original review Document **Sentiment: 1**	<s> i don't know how this place is even open anymore. <s> they've constantly put signs out front and even on the corner advertising deals practically giving their pizza away. <s> i gave this place a try when it first opened around the corner, but one try was more than enough. <s> when you 're trying to grab a quick bite to eat on the way home from work, this isn't the place to go. <s> even for a cold pizza and calzone, i waited 30 min... that's plain ridiculous. <s> the owner talked my ear off, not that that's a bad thing, but look dude, i wanted to be nice and support a local business, but once i left your shop and unfortunately tried your "food" (??) <s> i was done. **Predicted sentiment: 1**
Sentiment-preserving summary (20%)	<s> even for a cold pizza and calzone, i waited 30 min... that's plain ridiculous. **Predicted sentiment: 1**
Sentiment-preserving summary (50%)	<s> i don't know how this place is even open anymore. <s> even for a cold pizza and calzone, i waited 30 min... that's plain ridiculous. <s> the owner talked my ear off, not that that's a bad thing, but look dude, i wanted to be nice and support a local business, but once i left your shop and unfortunately tried your "food" (??) **Predicted sentiment: 1**

Table 4 Sentiment-preserving extractive summarization results with 20 and 50% summarization ratios on the exemplar review document from the Yelp dataset

Original review Document **Sentiment: 5**	<s> i lived around the corner from this place and must say i was somewhat skeptical that it would have staying power (considering how many other places in the whole food 's mall have closed after short runs). <s> i first visited here to pick up some peppercorns for a dinner that i was preparing. <s> i must say they have a great selection not only of peppercorns, but of most every other spice and herb . <s> i would highly recommend the dried peppers from all over the world and in levels of hotness . <s> penzey 's also has various herbal combinations including my two favorites - bangkok and singapore - the names described the flavors . <s> i would recommend purchasing in the glass jars vs. packages . <s> the packages contain way too much of the spice and/or herb and may dry out or lose flavor if sitting on a shelf too long . <s> i strongly suggest trying the blended combinations . <s> i keep some on hand when i have a tasteless meal that i need to liven up! **Predicted sentiment: 5**
Sentiment-preserving summary (20%)	<s> i would highly recommend the dried peppers from all over the world and in levels of hotness. **Predicted sentiment: 4**
Sentiment-preserving summary (50%)	<s> i lived around the corner from this place and must say i was somewhat skeptical that it would have staying power (considering how many other places in the whole food 's mall have closed after short runs) . <s> i must say they have a great selection not only of peppercorns, but of most every other spice and herb . <s> i would highly recommend the dried peppers from all over the world and in levels of hotness . <s> i keep some on hand when i have a tasteless meal that i need to liven up! **Predicted sentiment: 5**

Table 5 Sentiment-preserving extractive summarization results with 50% summarization ratios on the exemplar review document from the IMDB-10 dataset under models with and without the discriminator

Original review Document	<s> i grew up in the south and have spent time in the very section of the movie was set in and it so captured the mood and feel of the area.
Sentiment: 8	<s> i loved this film.
	<s> yes, it moves at a slower pace, it does n't feel the need to explain everything about everyone and what their backstory is.
	<s> that 's one of the things i so appreciated about it.
	<s> in this day and age where everything is simplified - everyone is a stock character, the plot goes from a to b to c and the end arrives with everything wrapped up and bad people have either immediately changed or been killed.
	<s> ah!
	<s> but this film is about real relationships and real feelings.
	<s> the performances are wonderful and adams deserves her oscar nomination.
	<s> the script is tight and smart.
	<s> the direction is dead-on.
	<s> the fact that a script like this can still find funding, get made and find an audience (no matter how small) is very encouraging.
	<s> hats off to junebug!
	Predicted sentiment: 8
Sentiment-preserving summary (50%) without discriminator	<s> yes, it moves at a slower pace, it does n't feel the need to explain everything about everyone and what their backstory is.
	<s> but this film is about real relationships and real feelings.
	<s> the performances are wonderful and adams deserves her oscar nomination.
	<s> the direction is dead-on.
	<s> the fact that a script like this can still find funding, get made and find an audience (no matter how small) is very encouraging.
	<s> hats off to junebug!
	Predicted sentiment: 8
Sentiment-preserving summary (50%)	<s> i grew up in the south and have spent time in the very section of the movie was set in and it so captured the mood and feel of the area.
	<s> yes, it moves at a slower pace, it does n't feel the need to explain everything about everyone and what their backstory is.
	<s> in this day and age where everything is simplified - everyone is a stock character, the plot goes from a to b to c and the end arrives with everything wrapped up and bad people have either immediately changed or been killed.
	<s> but this film is about real relationships and real feelings.
	<s> the direction is dead-on.
	<s> the fact that a script like this can still find funding, get made and find an audience (no matter how small) is very encouraging.
	Predicted sentiment: 8

eliminated. As shown in Table 5, while both model tend to select salient sentences preserving the main sentiments of the review, the result with a discriminator component is more natural. In particular, the sentence "in this day and age where everything is simplified - everyone is a stock character, the plot goes from a to b to c and the end arrives with everything wrapped up and bad people have either immediately changed or been killed." and the sentence "but this film is about real relationships and real feelings." are closely coupled, while the summary generated by the ablated variant only keeps the latter which begins with the conjunction "but", degrading the readability.

Human evaluation To better evaluate our proposed model, we perform human evaluation to judge the quality of the generated summaries. For each of the IMDB-10 dataset and the Yelp dataset, we randomly select 50 samples, and for each sample, we present

three extractive summaries with 50% summarization ratio, i.e., a randomly extracted summary, a summary generated by our proposed model, and a summary generated by the ablated variant of our model with the discriminator eliminated. We invited 3 human evaluators to assign two scores for each generated summary: 1) a sentiment-preserving score for the degree of the summary keeping the main sentiments of the original review; and 2) a readability score for how natural the summary is. Each score ranges from 1 to 3, where 1 indicates the lowest level and 3 indicates the highest level. The results are reported in Table 6 and Table 7. As shown in the results, the proposed model can achieve satisfying sentiment-preserving quality and readability under 50% summarization ratio, significantly surpassing the results from random extraction. While the ablated variant (i.e. without the discriminator) achieves similar, or slightly better, sentiment-preserving score with our model, it underperforms in terms of readability without any constraint on the coherences of the generated summary.

5.3.2 Sentiment classification accuracy

Table 8 reports the sentiment classification test accuracy on the IMDB-3, IMDB-10, and Yelp dataset with different summarization ratios, i.e., from 10 to 100%. For comparison, we include the sentiment classification accuracy for random sentence extraction for each summarization ratio.

We can see that, on all the datasets, the classification accuracy keeps increasing with more contents, while the improvement is diminishing after 50%. This does make sense since a model can make better prediction with more information. The interesting thing is that even with about half (or less, which is a trade-off between the sentiment classification accuracy and the summarization ratio) of the original document, the summarized version can achieve comparable accuracy with the full text (i.e. 100%). In contrast, the random sentence extraction underperforms the summaries generated with our proposed extractive SDS framework, especially for low summarization ratios. In particular, the difference of the sentiment classification performance between our extractive SDS framework and the random extraction is more significant on the IMDB-10 dataset than on the IMDB-3 and Yelp dataset. The reasons are two-fold: firstly, the 10-level classification is much more challenging and requires more details (which is easy to discard in a random selection method) than the 3-level and 5-level classification. Secondly, the movie reviews in IMDB dataset contain more sentiment-independent contents than restaurant reviews in the Yelp dataset (as shown in Table 3 and Table 4, most sentences can infer more or less the sentiment of the review), making random selected sentences more probable to predict the correct sentiment.

Table 6 Human evaluation results for extractive summaries with 50% summarization ratio on the IMDB-10 dataset

Model	Sentiment-preserving level	Readability level
Random	2.10	1.37
Proposed model w/o discriminator	2.79	2.07
Proposed model	2.75	2.46

Table 7 Human evaluation results for extractive summaries with 50% summarization ratio on the Yelp dataset

Model	Sentiment-preserving level	Readability level
Random	2.24	1.62
Proposed model w/o discriminator	2.81	2.29
Proposed model	2.77	2.53

We also provide the accuracy when we only train a sentiment classifier on top of a document encoder, i.e. without the sentence extractor. We can see that our framework will not harm the performance of the sentiment classifier, in fact it has a slight improvement, with the joint training with the sentence extractor.

5.3.3 Limitations

Although our proposed framework can achieve reasonable results in general, it still has some limitations on sentiment-preserving sentiment summarization. Firstly, it is unclear how to deal with the balance between the sentiment-preservation and summarization ratio. As shown in Table 8, lower summarization ratio can significantly degrade the sentiment analysis performance. Moreover, while the framework can generate summaries in different summarization ratios, there exists no good automatic metric to evaluate the summary in addition to the sentiment classification accuracy.

Secondly, with extractive summarization, it seems unavoidable to select the tedious sentence with only partial salient contents. In our future work, we plan to create a dataset suitable for strongly-supervised setting and abstractive summarization by inviting participants to generate ground-truth summaries.

Thirdly, in our problem statement in Section 3, we make the assumption that the sentiment of a sentiment-orientated document falls into one to the given classes. However, sometimes it can be much more complicated when the document possesses several different sentiment categories (e.g. with respect to different aspects of the concentrated object). In a restaurant review, a customer can be positive in terms of the cleanliness but negative in terms of the tastes. In such cases, intuitively we should deal with both the summarization component and the sentiment classification component in a finer level like aspects. We leave this for our future work.

Table 8 Sentiment classification accuracy with different summarization ratios

Summarization Ratios		10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	100% C-only
IMDB-3	SDS	73.6	77.3	79.9	81.2	82.0	82.3	82.6	82.7	82.8	83.1	82.8
	Random	64.2	68.2	72.4	75.1	77.1	78.6	79.5	80.7	81.5		
IMDB-10	SDS	38.8	42.5	44.7	45.7	46.5	47.1	47.5	47.7	47.8	48.1	47.9
	Random	24.8	28.6	32.3	35.5	38.5	40.3	42.0	43.5	44.9		
Yelp	SDS	51.7	55.6	59.3	62.2	63.9	65.1	65.8	66.4	66.8	67.1	66.9
	Random	42.3	46.7	51.4	55.4	58.2	60.3	62.4	64.0	65.1		

6 Conclusion and future work

In this paper, we have presented sentiment-preserving document summarization (SDS) to target at sentiment-orientated textual data such as reviews. Under the realistic weakly supervised setting where no desirable ground-truth summaries are available, we have devised an extractive summarization framework with the hierarchical document encoder, the sentence extractor, the sentiment classifier, and the discriminator. Experimental results on the IMDB movie review dataset and the Yelp business review dataset demonstrate that our framework can generate reasonable summaries with user-defined compression ratios.

Considering the limitations discussed in Section 5.3.3, for future work, we plan to create a labeled SDS dataset, so as to provide explicit supervision and enable the abstractive summarization which can generate more concise and natural summaries. Moreover, we intend to investigate finer sentiment granularity by integrating the techniques in the aspect-level sentiment analysis area.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Amplayo, R.K., Song, M.: An adaptable fine-grained sentiment analysis for summarization of multiple short online reviews. Data Knowl. Eng. **110**, 54–67 (2017)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proceedings of the International Conference on Learning Representations (2015)
3. Beineke, P., Hastie, T., Manning, C., Vaithyanathan, S.: Exploring sentiment summarization. In: Proceedings of the AAAI spring symposium on exploring attitude and affect in text: theories and applications, Vol. 39, pp. 1-4. (2004)
4. Cao, Z., Li, W., Li, S., Wei, F.: Improving multi-document summarization via text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 3053-3059. (2017)
5. Cheng, J, Lapata, M.: Neural summarization by extracting sentences and words. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (2016)
6. Conroy, J.M., O'Leary D.P.: Text summarization via hidden Markov models. In: Proceedings of Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 406–407. (2001)
7. Dai, A. M., Le, Q. V.: Semi-supervised sequence learning. In: Proceedings of the Neural Information Processing Systems, pp. 3079-3087. (2015)
8. Diao, Q., Qiu, M., Wu, C.-Y., Smola A. J., Jiang, J., Wang, C.: Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In: Proceedings of the International Conference on Knowledge Discovery and Data Mining, pp. 193-202. (2014)
9. Dieng, A. B., Wang C., Gao, J, Paisley, J.W.: TopicRNN: A Recurrent Neural Network with Long-Range Semantic Dependency. In: Proceedings of the International Conference on Learning Representations, 2017
10. Filatova, E., Hatzivassiloglou, V.: Event-based extractive summarization. In: Proceedings of the Meeting of the Association for Computational Linguistics Workshop on Summarization, pp. 104–111. (2004)
11. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y. Generative adversarial nets. In: Proceedings of the Neural Information Processing Systems, pp. 2672-2680. (2014)
12. Gu, J., Lu, Z., Li, H., Li, V.O.K.: Incorporating copying mechanism in sequence-to-sequence learning". In Proceedings of the Annual Meeting of the Association for Computational Linguistics (2016)
13. Hu, M., Li, B.: Mining and summarizing customer reviews. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 168-177. (2004)

14. Jean, S., Cho, K., Memisevic, R, Bengio, Y.: On using very large target vocabulary for neural machine translation". In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (2015)
15. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 655-665. (2014)
16. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1746-1751. (2014)
17. Kim, D.H., Ganesan, K., Sondhi P., Zhai, C.X.: Comprehensive review of opinion summarization. 2011
18. Kingma, D.P., Ba, O.: Adam: a method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations (2015)
19. Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R.S., Urtasun, R., Torralba, A., Fidler, S.: Skip-thought vectors. In: Proceedings of the Neural Information Processing Systems, pp. 3294-3302 (2015)
20. Kupiec, J., Pedersen, J.O., Chen, F.: A trainable document summarizer. In: Proceedings of Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 68–73. (1995)
21. Lerman, K., Blair-Goldensohn, S., McDonald, R.T.: Sentiment summarization: evaluating and learning user preferences. In: Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics, pp. 514-522. (2009)
22. Li, J., Chen, X., Hovy E. H., Jurafsky, D.: Visualizing and Understanding Neural Models in NLP. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 681-691. (2016)
23. Li, J., Monroe, W., Shi, T., Jean, S, Ritter, A., Jurafsky, D.: Adversarial learning for neural dialogue generation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 2157-2169. (2017)
24. Liu, P., Qiu, X., Huang, X.: Adversarial multi-task learning for text classification. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (2017)
25. Liu, L., Lu, Y, Yang, M., Qu, Q., Zhu, J., Li, H.: Generative Adversarial Network for Abstractive Text Summarization. In: Proceedings of the AAAI Conference on Artificial Intelligence (Abstract) (2018)
26. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J.R., Bethard, S., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (System Demonstrations), pp. 55-60. (2014)
27. Miao, Y., Blunsom, P.: Language as a latent variable: discrete generative models for sentence compression. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 319-328 (2016)
28. Mihalcea, R., Tarau, P.: TextRank: Bringing Order into Text. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 404-411. (2004)
29. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the Neural Information Processing Systems, pp. 3111-3119. (2013)
30. Mithun, S., Leila, K.: Summarizing blog entries versus news texts. In: Proceedings of the Workshop on Events in Emerging Text Types, pp. 1–8. (2009)
31. Mou, L., Peng, H., Li, G., Xu, Y., Zhang, L, Jin, Z.: Discriminative neural sentence modeling by tree-based convolution. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 2315-2325. (2015)
32. Nallapati, R., Zhou, B., dos Santos, C.N., Gülçehre, Ç, Xiang, B.: Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In: Proceedings of the SIGNLL Conference on Computational Natural Language Learning, pp. 280-290. (2016)
33. Nallapati, R, Zhai, F, Zhou, B.: SummaRuNNer: a recurrent neural network based sequence model for extractive summarization of documents. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 3075-3081. (2017)
34. Nenkova, A., Vanderwende, L., McKeown, K.: A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In: Proceedings of Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 573–580. (2006)
35. Paulus, R., Xiong, C, Socher, R.: A deep reinforced model for abstractive summarization. CoRR abs/1705.04304, (2017)
36. Radev, D. R., Allison, T., Blair-Goldensohn, S., Blitzer, J, Çelebi, A, Dimitrov, S, Drábek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., Zhang, Z.: MEAD - a platform for multidocument multilingual text summarization. In: Proceedings of the International Conference on Language Resources and Evaluation (2004)

37. Rennie, S.J., Marcheret, E., Mroueh, Y, Ross, J, Goel, V.: Self-critical sequence training for image captioning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1179-1195. (2017)
38. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 379–389. (2015)
39. See, A., Liu, P.J., Manning, C.D.: Get to the point: summarization with pointer-generator networks. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (2017)
40. Sutton, R.S., McAllester, D.A., Singh, S.P., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. In: Proceedings of the Neural Information Processing Systems, pp. 1057-1063. (1999)
41. Tai, K.S., Socher, R, Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 1556-1566. (2015)
42. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1422-1432. (2015)
43. Titov, I., McDonald, R.T.: A joint model of text and aspect ratings for sentiment summarization. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 308-316. (2008)
44. Woodsend, K., Lapata, M.: Automatic generation of story highlights. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 565–574. (2010)
45. Xu, W., Sun, H., Deng, C., Tan, Y.: Variational Autoencoder for Semi-Supervised Text Classification". In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 3358-3364. (2017)
46. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480-1489. (2016)
47. Yang, Z., Chen, W., Wang, F., Xu B.: Improving neural machine translation with conditional sequence generative adversarial nets. CoRR abs/1703.04887 (2017)
48. Zhang, X., Zhao, J.J., LeCun, Y.: Character-level convolutional networks for text classification. In: Proceedings of the Neural Information Processing Systems, pp. 649-657. (2015)