Topic Segmentation for Dialogue Stream

Leilan Zhang[†], Qiang Zhou[*]
[†] Department of Computer Science and Technology, Tsinghua University, Beijing, China
E-mail: zll17@mails.tsinghua.edu.cn
[*] CSLT, Research Institute of Information Technology, Tsinghua University, Beijing, China
E-mail: zq-lxd@mail.tsinghua.edu.cn

Abstract—Topic segmentation, which aims to divide a document into topic blocks, is a fundamental task in natural language processing. Most of the previous researches focus on written text rather than dialogue text. However, dialogue text has its unique characteristic and is more challenging in topic segmentation. The existing neural models for topic segmentation are usually built on RNN or CNN, which are competent in written text but has a poor performance in dialogue text. We argue that a better segmentation result for dialogue text requires a better semantic representation of sentences. In this paper, we formulate topic segmentation as a sequence labeling task and propose a model based on BERT and TCN (Temporal Convolutional Network) to accomplish the task. We also present three datasets, including two dialogue datasets and a news dataset, to evaluate the model's performance. Compared to the previous best model, our model shows an absolute performance improvement of 8% - 17% in F_1 scores. Moreover, we explore the impact of importing speakers on dialogue text segmentation, the experiment result shows that the additional speaker information could effectively improve the segmentation performance.

I. Introduction

Topic segmentation is defined as dividing a document into multiple segments according to their topics. It plays an important role in natural language processing and provides the foundation for tasks like text summarization, information retrieval, dialogue analysis, and etc. For example, in the case of finding a particular part in a long text (like a meeting record or a subtitle), it is hard to locate the start of the interested segment unless reading the whole document. However, it would be much easier to retrieval if the document is organized as topic segments.

Multiple models have been proposed for topic segmentation, including supervised and unspervised methods. However, most of these methods are focus on written text (such as medical textbooks[1], fiction novels[2], wikipedia[3] and etc.), the lack of research on dialogue text is a surprising fact. Generally, written texts already have their structure characteristic. For example, the topic blocks of a textbook is organized as paragraphs, and there might be captions to prompt the start of new topics. Written texts are usually composed of long sentences, which contains many words related to the topics. Compared with the written texts, dialogue texts commonly come in the form of stream data, which has no specific structure like captions or paragraphs, and the utterances of dialogue texts are usually shorter in length and more obscure to reflect the topics. Therefore, topic segmentation task in dialogue texts is more challenging than that in written texts [4].

In recent years, neural network models are applied for topic segmentation and achieve better results than previous supervised methods. They are mainly employed in two ways: to treat the problem as a sequence labeling task or to treat the problem as a sentence-pair classification task. Although the existing methods do well in segmentation in written text, their performances are still poor in segmenting dialogue text. We argue that this problem is due to the poor capability to represent the semantics of sentences and to capture the global information of the dialogue text.

Therefore, we formulate the topic segmentation problem as a sequence labeling task and divide the process into two steps: 1) learn the representation of the sentences and encode them as vectors; 2) detect topic conversion through the vectors. In step 1, we use BERT [22] in our model to encode sentences. In step 2, we adopt the Temporal Convolutional Network [23] to detect the topic conversion.

We provide three datasets to evaluate the proposed model: DAct, Sub, and Weibo. Both DAct and Sub are dialogue texts, while Weibo is written texts. We also compare the performance of the proposed model with the competitive models on the datasets. Experiment results show that our model achieved the highest F_1 scores on all three tasks (Weibo: 0.9, DAct: 0.81, Sub: 0.71).

In summary, the contributions of this paper include the following:

- We propose a topic segmentation model for dialogue text based on BERT and TCN, which achieves the best results on both news and dialogue texts.
- We conduct experiments on dialogue datasets DAct and Sub and show that the speaker information can effectively improve the segmentation performance.
- We provide three datasets, including news and dialogue text, which can facilitate researches in the field of topic segmentation, text summarization, and etc.

The rest of the paper is organized as follows. Section 2 reviews the development of topic segmentation and language representation. Section 3 illustrates the architecture of our model in detail. Section 4 describes the construction of the datasets, the evaluation metrics and the experiment results and analysis. We present a brief conclusion in Section 5.

II. Related Work

A. Topic Segmentation

Methods for topic segmentation include unsupervised and supervised methods.

Unspervised methods can be classified into two categories: methods based on similarity and probabilistic generative methods[5]. The similarity-based methods assume that sentences in the same segment are more similar than those in different segments. Therefore, the topic conversion can be detected by the change of similarity between adjacent sentences. The representive models are Texttiling[6], C99[7], LCSeg[8]. The probabilistic generative methods assume that a document consists of a sequence of hidden topic variables, and each topic has its own distribution. Therefore, the topic conversion can be speculated through changes in word distribution. Ref [4] proposed PLDA, which computes the amount of common topic distribution between two adjacent paragraphs. Ref [9] and [10] proposed methods to calculate sentence similarity based on LDA [11].

Supervised methods include classifier based on decision trees[12] and probabilistic models[13][14]. Ref [15] combines lexical features (like lexical similarity) and conversational features (like long pauses, speaking rates shifts, silence and etc) for topic and sub-topic segmentation. Ref [16] integrates lexical and syntactic features in a segmentation model based on CRF. Supervised methods usually have better performance than unsupervised methods, but they depend on a huge amount of labeled data and handcrafted features.

In recent years, some researchers explore the application of neural network methods to topic segmentation tasks. Ref [17] proposes a sequence labeling architecture based on BiLSTM and CNN for topic segmentation for the first time. Ref [18] proposes a model based on CNN to rank the semantic coherence through learning the partial ordering relations among paragraphs. Ref [3] uses CNN and BiLSTM to encode sentences and contexts respectively and import an attention mechanism to solve the BiLSTM's long-range dependency problem. The segmentation task is completed by classifying whether the current sentence is a topic conversion point or not. The model based on RNN is proposed in [19] for segmentation on transcripts generated by speech recognition. Ref [20] constructs a segmentation model based on two layers of BiLSTM, the low-level layer encodes the semantics of sentences while the high-level layer encodes the context information. However, BiLSTM is not good at processing long-range dependency, which makes its semantic representation of a sentence not as good as that of the Transformer. Moreover, BiLSTM cannot compute in parallel, which makes it require lots of time for training. Therefore, we employ TCN in our model, which is good at sequence modeling and can compute in parallel, to detect the topic conversion.

B. Language Representation

Generally, when applying the neural network method in various NLP tasks, a sentence is first tokenized and represented as a matrix $X = (x_1, x_2, \cdots, x_n)$, in which x_i denotes the ith token's embedding vector. The sentence matrix is then used as input of the model.

Different embedding methods have different abilities of the semantic representation. The Word2Vec proposed in [24] can capture the semantic relations between words and is helpful in many NLP tasks. However, instead of adjusting a word's embedding according to its context, Word2Vec can only map a word to a fixed vector statically, therefore, it cannot distinguish the different meanings of a polysemy. To solve this problem, ELMo proposed in [25] imports two 2-layer LSTMs trained with both forward and backward language models and uses different layers to capture the syntactic and semantic features of the sentence respectively. When encoding a sentence, the original word vector is summed with vectors of the other two layers, so that the word representation is no longer a fixed vector, but can be dynamically adjusted according to its context. ELMo achieved the highest scores in 6 NLP tasks. To improve the feature extraction ability of the network, instead of using LSTM, GPT adopts transformer[21] as a feature extractor and achieves the best results in 9 NLP tasks[26]. However, GPT only uses the forward language model for training, which makes GPT can only predict according to the left part. BERT improves this structure by the adoption of multi-layer Transformers and the bidirectional language model so it can combine both left and right part to do predictions, which makes it achieves the best results in 11 NLP tasks[22]. Therefore, in order to improve the quality of sentence representation, we employ BERT to embed sentence to its semantic feature vector.

III. Method

In this section, we will give a formal definition of the topic segmentation task and a detailed description of the proposed model's architecture.

As mentioned above, there are two main architectures for topic segmentation neural models. One is to introduce the context in a window of size k and classify whether the current sentence is the topic segment boundary or not. The other is to label a sequence of sentences to indicate the topic conversion points. The classification architecture can only capture the local information, while the sequence labeling architecture can grasp the global structure of the document. Therefore, we formulate the topic segmentation problem as a sequence labeling task as following:

- Input: a scene segment of length M, which contains utterances $\{S_1, S_2, \cdots, S_M\}$ and is composed of several topic segments $T_1, T_2, \cdots, T_N$. Each topic segment is related to some topic and contains a few utterances. The scene segment is a segment from the dialogue stream and is obtained with some preprocess steps.

- Output: a label sequence of the same length M: $\{y_1, y_2, \cdots, y_M\}$, where $y_i \in \{0, 1\}$ indicates whether S_i is the start of a new topic segment.

Our topic segmentation model consists of two main parts: sentence representation and topic segmentation. The sentence representation module is used to embed sentences as vectors; the topic segmentation module receives those vectors and detects topic segment boundaries. The representation module in our model will encode sentences to their vectors based on BERT.

A. Sentence Representation

BERT is a multi-layer bidirectional network, in which each layer is a Transformer. Given a sentence $[w_1, w_2, \cdots, w_m]$, E_i denotes w_i's input representation, which is constructed by summing the corresponding token, segment, and position embeddings. BERT provides a model of Chinese with hyperparameters of L=12, H=768, A=12, where L is the number of layers (i.e. the number of Transformers), H denotes the size of the hidden layers, and A denotes the number of self-attention in the Multi-Head Attention. BERT is pre-trained on a large text corpus by performing the "masked LM" and the "next sentence prediction" tasks. BERT uses character-based tokenization for Chinese. Therefore, given a sentence containing N words, BERT will output a feature vector of size H for every single character, and the entire sentence will be represented as a matrix of $N * H$. In the inner part of BERT, each layer adds self-attention to the output of the previous layer and outputs a tensor of shape $[N, H]$.

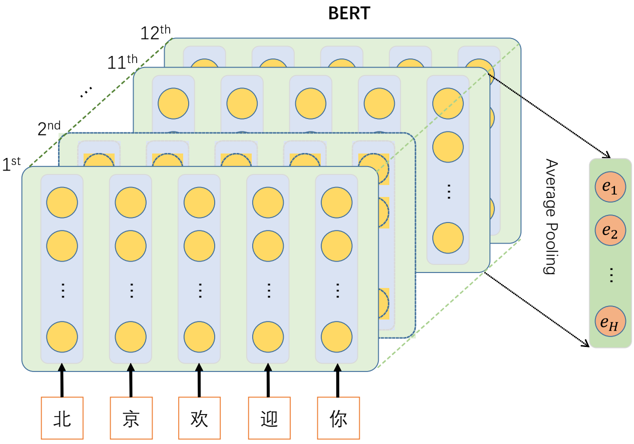

Fig. 1. Extract Sentence Representation from BERT

Like multi-layer LSTM, the weights in the high layer of the network usually contain task-related information. As shown in [28] [29], the top layers of a network contain high-level information while the bottom layers contain low-level features, therefore, in the transfer learning methods, the high-level layers are usually dropped and only the low-level features are adopted to feed into the downstream models. Since BERT is pre-trained for the "masked LM" and the "next sentence prediction" tasks, the closer to the last layer, the weights are more biased to the two targets; the closer to the first layer, the weights are more similar to the original word embedding but contains less high-level semantic information. Therefore, considering both semantic representation ability and computation complexity, we extract output tensor from the second-to-last layer of BERT, and apply the average pooling strategy to it to generate a vector of size $[H]$ as the feature vector of the input sentence. Fig 1 displays the process.

B. Topic Segmentation

The architecture of bidirectional LSTM with CRF is usually adopted to accomplish sequence labeling tasks [27][3]. However, LSTM has problems in dealing with long-range dependency in practice. For a long document, LSTM is not good at grasp its global structure. Moreover, LSTM cannot compute in parallel and is slow in convergence. Therefore, we choose TCN rather than LSTM as the topic segmentation module in our model[23].

TCN is proposed for sequence modeling tasks. For sequence data of size N, TCN will produce the prediction sequence of the same size. The most notable characteristic of TCN is the dilated convolution. It can ensure that every hidden layer of TCN has the same size as the input sequence and the receptive field is larger than that of a 1-D CNN with the same number of layers. TCN uses causal convolution to ensure that the prediction of time step t will only rely on the information before time step $t-1$ and there's no information "leakage" from future to past. This property is very suitable for our task since a dialogue stream develops in chronological order. The residual convolution is also used in TCN so that the features from bottom layer can be sent to the top layer directly to improve the network's performance. These properties enable TCN to learn the overall structure of the sequence better. In addition, compared with RNN, TCN can be computed in parallel, which greatly improves the training and predicting speed.

For the sentence sequence $\{S_1, S_2, \cdots, S_M\}$, we use BERT as encoder to obtain their semantic representation $\{E_1, E_2, \cdots, E_M\}$. These sentence vectors are then fed into the TCN to output a label sequence of 0-1, where label 1 indicates the boundary of the topic segment. The overall architecture of the segmentation model is shown in Fig 2. It is noteworthy that the BERT part is seperated from the TCN part in our model for the reason of computing efficiency, therefore, the backpropagation process will only update the parameters of the TCN part.

Due to the sparsity of topic boundaries, the distribution of 0 and 1 in the label sequence is extremely unbalanced, which tends to cause bias of the model. To solve the problem of imbalanced class, we use Focal Loss[30] as the loss function to optimize TCN. The formula of the Focal Loss is shown below:

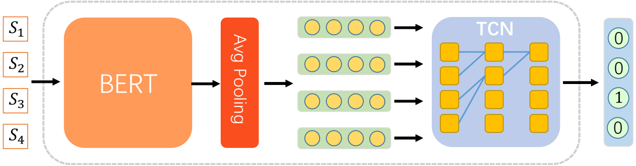

Fig. 2. Architecture of the Topic Segmentation Model

$$L_{fl} = \begin{cases} -(1-\hat{y})^{\gamma} log\hat{y}, \ if \ \ y=1 \\ -\hat{y}^{\gamma} log(1-\hat{y}), \ if \ \ y=0 \end{cases} \tag{1}$$

Where γ is a positive number and usually set to 2. The starting point of Focal Loss is to use different penalty scores to adjust the attention to different classes. For example, when the negative samples ($y = 0$) in the data set are much more than the positive samples ($y = 1$), the model tends to classify a sample as a negative one ($\hat{y} = 0$). When $y = 0$ (negative sample), both $\hat{y}^{\gamma}$ and $log(1-\hat{y})$ are small, the model doesn't need to adjust too much on these samples; when $y = 1$ (positive sample)), both $(1-\hat{y})^{\gamma}$ and $log\hat{y}$ are very large, the model needs to adjust a lot on these samples. Therefore, the originally small number of positive samples will have a greater impact on the model, which solves the class imbalance problem effectively.

IV. Experiment and Analysis

In this section, we will introduce the construction process of the datasets, the metrics for evaluation, experimental results and analysis, and several improvement measures for the existing model.

A. Data Preparation

We present three data sets: Weibo[1], DAct [31], and Sub[2], where Weibo is short news collected from social media, DAct is two-participants dialogue text, and Sub is a multi-participants dialogue text extracted from subtitles. The Weibo dataset is used as a reference to evaluate our model's performance on written text, and we assume that each Weibo news is about the same topic, so a piece of news can be considered as a topic segment. For DAct and Sub datasets, we manually label the topic boundaries in the conversations. Thus, the original topic segments of Weibo, DAct and Sub are obtained and there are topic conversions between two topic segments.

Due to the small scale of the original dataset and the high cost of manual annotation, we use a simple augmentation strategy to automatically construct and expand training datasets. The topic segments are randomly selected and concatenated to form a new scene segment, in which the concatenation points are the topic conversion points. To be specific, the original weibo dataset, subtitle dataset and dialogue action dataset has 12000, 3000 and 936 manually labeled topic segments, respectively. We randomly select topic segments from the original dataset, then concatenate the topic segments into a new scene segment. To imitate the original scene segment data, the constructed scene segment is concatenated by 3 or 4 or 5 topic segments, the amount of which is in a ratio of 1:1:1. With this augmentation strategy, a large amount of training data can be automatically generated. The basic statistics of the augmented datasets is listed in TABLE I.

TABLE I
Basic Statistics on Datasets

	Weibo	Sub	DAct
Num of scene segments (trainset)	20000	20000	20000
Num of scene segments (testset)	4000	4000	4000
Mean of scene segments' lengths	12.97	20	26.08
Std of scene segments' lengths	3.9	5	10
Mean of utterances' lengths	23.75	9.39	10.24
Std of utterances' lengths	11.99	4.64	4.94

We construct 24000 scene segments for each dataset and split them into trainset and testset in the ratio of 5:1. From TABLE I, we can observe that the average length of utterances in Weibo is larger than that in Sub and DAct, which indicates that sentences of Weibo contain more words than those in Sub and DAct. This difference may have impact on the segmentation results since the long sentence could contain more words related to the topics than the short ones. Fig 3 shows a segment randomly selected from the Sub dataset. The coloumn "id" is the index of every utterances. The column "spkr" is set to 1 if the speaker of current utterance is different from the previous one's, otherwise, it is set to 0. For example, the "spkr" label of utterance at id 2 is 1, which means the speaker of this utterance is different from the utterance at id 1; the 'spkr' label of utterance at id 10 is 0, which means this utterance has the same speaker with utterance at id 9. The column "content" displays the utterance content and column "ref" is the reference label to indicate whether the corresponding utterance is the topic conversion point. For example, the first segment of id 0-8 is related to medicine, while the subsequent segment of id 9-14 is related to detectives, and there is an obvious topic conversion between the two segments, so the "ref" label at id 9 is set to 1. The column "pred" is the prediction result of our model (this column doesn't appear in the dataset).

B. Evaluation Metrics

We use two metrics: F_1 score and WinDiff[32] to evaluate the performance of the proposed segmentation model on the utterance level. Moreover, we propose a "Span" metrics to measure the model's performance on the segment level.

1) F_1 and WinDiff: Given the reference segmentation R, the F_1 score only focuses on the segmentation points, it is defined as $\frac{2*p*r}{p+r}$, where p is the precision rate of

[1] https://news.sina.com.cn
[2] http://assrt.net

id	spkr	content	ref	pred
0	1	他还活着吗? He still alive?	0	0
1	1	是的 Yeah	0	0
2	1	给他用肝素，静脉推注免疫球蛋白 Start him on heparin and iv immunoglobulin.	0	0
3	1	治疗恶性萎缩性丘疹? For degos?	0	0
4	1	心脏停搏并不是那种恶心的停搏 The cardiac arrest wasn't just a disgusting arrest.	0	0
5	0	是冠脉病变，冠状动脉是大血管 It was a coronary event. Coronaries are large vessels.	0	0
6	0	也就是说，这不是恶性萎缩性丘疹 Mean's it can't be degos.	0	0
7	1	但是组织活检证实... But the biopsies confirmed ···	0	0
8	1	亚瑟・柯南・道尔也曾是秘密会员 Sir Arthur Conan Doyle was a secret member.	1	1
9	1	而且，只是传说 And ···it's a myth	0	0
10	0	是老私家侦探告诉菜鸟的童话 It's a fairy tale old P.I.s tell rookies	0	0
11	0	好骗他们干苦活儿 to get them to do the scut work.	0	0
12	1	你就是吃不着葡萄说葡萄酸 Said the detective that didn't get an invitation.	0	0
13	1	好吧，不管真假 All right, real or not	0	0
14	1	那个酒吧认识的男的呢 What about the guy from the bar?	1	1
15	0	你给他电话号码的那一个 The guy you gave your number to.	0	1
16	1	你是怎么知道的 How do you know about that?	0	0
17	1	因为他打来找你 Because he called here looking for you.	0	0
18	0	所以别跟我说，你只是亲了一次男同事 So don't tell me kissing this guy is a one-time thing.	0	0
19	0	因为你又在酒吧，又在阳台上 Because you've been out in bars and on balconies	0	0
20	0	已经一个多月了 for over a month now.	0	0
21	0	你都没有礼貌性和我说一声 And you don't even have the courtesy to tell me.	0	0
22	1	为什么我没有拿到留言 Why didn't I get that message?	0	0
23	1	但我总觉得有些尴尬 I felt really awkward about the whole thing.	1	0
24	1	别这么想 Don't worry about it.	0	1
25	0	这身是奶奶的茶壶套吗 Is this grandma's tea cozy?	0	0

Fig. 3. Segmentation result on dialogue from Sub

prediction of label 1 and is defined as the ratio of true label 1 in the predicted label 1, r is the recall rate of label 1 and is defined as the ratio of predicted label 1 in the reference label 1. WinDiff introduces a sliding window of size k to compare the predicted segmentation H with R, where k is usually set to half of the average length of the segments in R.

WinDiff is defined as:

$$WinDiff = \frac{1}{N-k}\sum_{i=0}^{N-k}(|R_{i,i+k} - C_{i,i+k}| \neq 0) \qquad (2)$$

Where $R_{i,i+k}$ is the number of reference boundaries from position i to $i+k$ in the window, and $C_{i,i+k}$ is the number of predicted boundaries in the same window. As a probabilistic metrics for measuring segmentation errors, the values of WinDiff is between 0 and 1. The smaller the values of WinDiff, the closer the segmentation H is to the segmentation R. When they are identical to each other, WinDiff equals 0.

2) Span: In order to measure the model's performance on segment level and intuitively describe the segmentation results on the dialogue texts, we introduce the Span evaluation metrics.

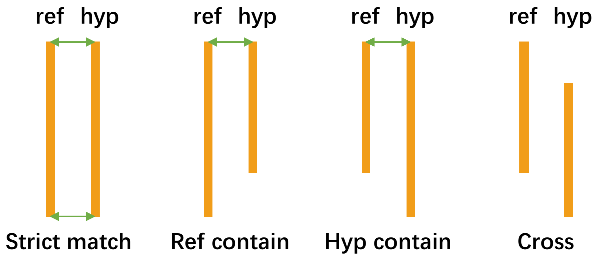

Fig. 4. Four cases of Span metrics

As shown in the Fig 4, we compare the two segments generated by the reference segmentation (”ref”) and the hypothesized segmentation (”hyp”) at the same location. There are four cases:

- I. Strict match: the starting and ending points of ref and hyp are exactly the same;
- II. Ref contain: ref and hyp have only one segmentation point, and hyp is shorter so that it is contained by the ref, which indicates that the model classifies a position as a topic conversion point while it's actually not;
- III. Hyp contain: ref and hyp have only one same segmentation point, and hyp is longer so that it contains ref, which indicates that the model misses the segmentation point;
- IV. Cross: the start and end points of ref and hyp are neither different, but there are intersection parts, which is a serious segmentation error.

The Span metrics is determined by the proportion of these four categories in the results.

TABLE II
Experiment results on the test sets

	Weibo		DAct		Sub	
	$WinDiff$	F_1	$WinDiff$	F_1	$WinDiff$	F_1
2L BiLSTM	0.2770	0.82	0.2962	0.675	0.3237	0.544
BERT+TCN	0.1267	0.90	0.1957	0.81	0.2408	0.71

TABLE III
Evaluation of the improved model

	DAct			Sub		
	F_1	$Precision$	$Recall$	F_1	$Precision$	$Recall$
baseline(BERT+TCN)	0.81	0.85	0.77	0.71	0.73	0.69
+Speaker	0.86	0.90	0.82	0.81	0.82	0.81
+CRF	0.82	0.87	0.77	0.75	0.81	0.69
+Speaker +CRF	0.86	0.91	0.82	0.82	0.84	0.81

TABLE IV
Statistics on the segmentation result with or without speaker

	reference≠prediction				reference=prediction			
	r0p1s0	r0p1s1	r1p0s0	r1p0s1	r0p0s0	r0p0s1	r1p1s0	r1p1s1
No Speaker	1584	1532	156	3633	35408	29378	283	8026
+ Speaker	4	2135	436	1899	36988	28775	3	9760
diff	-1580	+603	+280	-1734	+1580	-603	-280	+1734

C. Experiment Results

We compared the performance of the proposed model with the previous best model (BiLSTM [20]) on the three datasets. From Table II, our model achieves the best results on all the datasets. Compared to BiLSTM, the F_1 scores on the Weibo, DAct, and Sub datasets increased by 0.08, 0.135, and 0.166, respectively. WinDiff has a certain degree of reduction.

On the other hand, the performances of our model on the three datasets are quite different. The F_1 score on the Weibo data achieves 0.9, while the F_1 on the Sub data is only 0.71, which indicates the complexity of topic segmentation varies with the type of document. Since the Weibo dataset consists of news texts, which are composed of long sentences and vocabularies that are associated with their topics. In contrast, the Sub dataset is mainly dialogue texts, which are composed of shorter utterances and colloquial words that are obscure with their topics, therefore, it is more difficult to detect the topic segment boundaries in dialogue texts.

TABLE V shows our model's Span metrics of the result on the three datasets. It can be observed from TABLE V that the proportion of case I has a positive correlation with the F_1 score of the model. This shows that the Span metrics is compatible with the F_1 score and is well-defined, a high proportion of case I means a close distance between the hyp and the ref segmentation, while a high proportion of case IV (cross) means a poor segmentation performance of the model.

It can be observed that the distribution of the case I proportion in the Span metrics is consistent with that of the F_1 score on the datasets, which is the highest on the Weibo and the lowest on the Sub. In addition, the ratio of Ref contain and Hyp contain is very close in each task, indicating that the segmentation error distribution of the model is uniform, and the preference of over-segmentation and over-conservation is small.

TABLE V
The Span Score on Three datasets

	Strict	Ref Contain	Hyp Contain	Cross
Weibo	0.7999	0.0979	0.1005	0.0015
DAct	0.6438	0.1549	0.1967	0.0046
Sub	0.4729	0.2757	0.2383	0.0130

D. Improvement Measures

Unlike Weibo, the dialogue datasets Sub and DAct also contain speaker information. As displayed in Fig 3, several consecutive utterances may belong to the same speaker (such as utterances with the id of 9, 10, and 11), and speaker information may have a great influence on the segmentation result. In addition, as it shows, the model predicts two continuous utterances (id of 14 and 15) both as segmentation points, while there should be only one actually (before the utterance with id 14). If such errors could be filtered out, the segmentation result should be more accurate. Therefore, we introduce speaker labels and add a CRF layer after the output of the TCN. The speaker label is set to 1 if the speaker of current utterance is different from the previous one's, otherwise, it is set to 0.

From TABLE III, it can be observed that after introducing the speaker information, the F_1 scores on DAct and Sub are increased by 0.05 and 0.1, respectively, both precision and recall rate are greatly improved. TABLE VI displays the statistics on ref-speaker labels in the test set of Sub data. The notation r0s1 means the number of utterances whose reference label and speaker label is 0 and 1 respectively. It can be observed that the proportion of r1s0 is relatively small (439/80000), which means the topic conversion is usually less likely to happen if the speaker doesn't switch.

TABLE VI
Statistics of Ref-Speaker Label

r0s0	r0s1	r1s0	r1s1
36992	30910	439	11659

TABLE IV compares the segmentation results before and after adding speaker information. The notation r0p1s0 denotes utterances whose reference label, prediction label, and speaker label is 0,1,0 respectively, it indicates that the model predicts there's a topic conversion at this utterance although there's actually not, and the speaker of the current utterance is the same as the previous one's.

Watch the diff row of TABLE IV, a positive difference under "ref≠pred" title (or "ref=pred") means a performance reduction (or improvement) while negative difference under "ref≠pred" title (or "ref=pred") means a performance improvement (or reduction). It can also be observed that a column has a positive difference if and only if $i = j$ in $r_x p_i s_j$; and a column has a negative difference if and only if $i \neq j$ in $r_x p_i s_j$. This means that the prediction of the model has the same trend with the speaker labels. For example, after introducing speaker information, r0p1s0 decreases from 1584 to 4, meaning the frequency of predicting an utterance as the topic boundary reduces 1580 if the speaker label shows there's no speaker switch; r0p1s1 increases from 1532 to 2135, meaning the frequency of predicting an utterance as the topic boundary increases 603 if the speaker label shows there's a speaker switch although there's no topic conversion. This indicates that the speaker-switching information could improve the probability of segmenting while the speaker-holding information could decrease that. So the speaker information in fact forms a constraint of the position of the segmentation point.

This constraint could bring some unexpected results. For example, r1p1s0 decreases from 283 to 3, indicating that the model predicts the topic conversion correctly when there's no speaker tags while it fails to detect those topic conversions when the speaker tags exist, which show there's no speaker switch. This strange phenomenon is due to the errors produced in the data construction process. Fig 5 shows a snippet from Sub, the utterances from id 11 to id 15 are all spoken by the character "Castle", however, the utterances with id of 11 to 12 are from one topic segment while the utterances with id of 13 to 15 are from another. Therefore, although these utterances are belong to the same speaker, they are not from the same scene and there are indeed a topic conversion among them. Then the lack of speaker-switching label makes the model wrongly judge the topic boundaries, which results in a performance reduction in such case. Fortunately, this kind of error only takes a small proportion in the whole dataset (nearly 280/20000=1.4%) and it also verifies that the model tends to segment where the speaker changes.

id	spkr	content	ref	pred
10	Director	非常不错大伙们 Very nice	0	0
11	Castle	有没有什么特殊的人 Was there anyone special in his life?	1	1
12	Castle	他倒想呢但是没有 Oh, he wished, but no.	0	0
13	Castle	将其放大测试 have it amplified, tested.	1	0
14	Castle	而且我肯定你们的证据存储科 And I'm sure your evidence storage	0	0
15	Castle	完好的保留了你的海豚 has kept your dolphin well preserved.	0	0
16	Chief	泰迪 Teddy	0	0

Fig. 5. A wrong predicted example

Due to the fact that topic conversion is more likely to happen when the speaker switches, introducing speaker information could result in a more accurate segmentation. After adding the CRF layer, the precision of the model increases by 0.02-0.08, while the recall rate isn't changed, indicating that the CRF layer does filter out some unreasonable predictions. Combining the two methods, the F_1 scores of the model on DAct and Sub are increased by 0.15 and 0.11 respectively, demonstrating the effectiveness of the improvement measures.

V. Conclusions

Since the existing topic segmentation methods do not work well on the dialogue text, we formulate the topic segmentation problem as a sequence labeling task and propose a model based on BERT and TCN, in which BERT is used for sentence representation and TCN is used for topic conversion detection. Compared with the previous best model, the proposed model achieves the better result on both written texts (Weibo) and dialogue texts (Sub and DAct). We also propose several improvement measures for the dialogue text. The experiment shows that the introduction of speaker information can effectively improve the precision of the topic segmentation on dialogue. However, the current model uses the default weights of BERT to extract sentence representation. In future work, we will fine-tune the model on the domain-specific text before embedding the sentences and explore the influence of different embedding methods on the topic segmentation.

VI. Acknowledgement

This work has supported by grants from National Natural Science Foundation of China (No.61433018, No.61373075). We would also like to thank the administrator of website http://assrt.net for the help of providing the mirror of their subtitle data.

References

[1] Eisenstein, Jacob and Barzilay, Regina. "Bayesian unsupervised topic segmentation." Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP, pages 334–343, 2008.

[2] Kazantseva, Anna and Szpakowicz, Stan. "Linear Text Segmentation Using Affinity Propagation." Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 11, pages 284–293, 2011.

[3] Pinkesh Badjatiya, Litton J. Kurisinkel, Manish Gupta, and Vasudeva Varma. "Attention-based neural text segmentation." Advances in Information Retrieval, pages 180–193, Cham, 2018. Springer International Publishing.

[4] Matthew Purver, Thomas L. Griffiths, Konrad P. Körding, and Joshua B. Tenenbaum. "Unsupervised topic modelling for multi-party spoken discourse." Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, ACL-44, pages 17–24, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[5] Jing Li, Aixin Sun, and Shafiq Joty. "Segbot: A generic neural text segmentation model with pointer network." Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, pages 4166–4172. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[6] Marti A Hearst. "Texttiling: Segmenting text into multi-paragraph subtopic passages." Computational linguistics, 23(1):33–64, 1997.

[7] Freddy YY Choi. "Advances in domain independent linear text segmentation." arXiv preprint cs/0003083, 2000.

[8] Michel Galley, Kathleen McKeown, Eric Fosler-Lussier, and Hongyan Jing. "Discourse segmentation of multi-party conversation." Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1, pages 562–569. Association for Computational Linguistics, 2003.

[9] Hemant Misra, François Yvon, Joemon M. Jose, and Olivier Cappe. "Text segmentation via topic modeling: An analytical study." Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM '09, pages 1553–1556, New York, NY, USA, 2009. ACM.

[10] Martin Riedl and Chris Biemann. "Topictiling: a text segmentation algorithm based on lda." In Proc. ACL '12 Student Research Workshop, ACL '12, pages 37–42, 2012.

[11] Blei, David M and Ng, Andre Y and Jordan, Michael I "Latent dirichlet allocation." Journal of machine Learning research, vol 3, Jan, pages 993–1022, 2003.

[12] Gökhan Tür, Dilek Hakkani-Tür, Andreas Stolcke, and Elizabeth Shriberg. "Integrating prosodic and lexical cues for automatic topic segmentation." Computational linguistics, 27(1):31–57, 2001.

[13] Jeffrey C. Reynar. "Statistical models for topic segmentation." Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, ACL '99, pages 357–364, Stroudsburg, PA, USA, 1999. Association for Computational Linguistics.

[14] Masao Utiyama and Hitoshi Isahara. "A statistical model for domain-independent text segmentation." Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, 2001.

[15] Pei-Yun Hsueh, Johanna D. Moore, and Steve Renals. "Automatic segmentation of multiparty dialogue." 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006.

[16] Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. "A sequential model for discourse segmentation." Proceedings of the 11th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing'10, pages 315–326, Berlin, Heidelberg, 2010. Springer-Verlag.

[17] Liang Wang, Sujian Li, Xinyan Xiao, and Yajuan Lyu. "Topic segmentation of web documents with automatic cue phrase identification and blstm-cnn." Natural Language Understanding and Intelligent Applications, pages 177–188. Springer, 2016.

[18] Liang Wang, Sujian Li, Yajuan Lv, and Houfeng WANG. "Learning to rank semantic coherence for topic segmentation." pages 1340–1344, 2017.

[19] Imran Sehikh, Dominique Fohr, and Irina Illina. "Topic segmentation in asr transcripts using bidirectional rnns for change detection." 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pages 512–518. IEEE, 2017.

[20] Omri Koshorek, Adir Cohen, Noam Mor, Michael Rotman, and Jonathan Berant. "Text segmentation as a supervised learning task." Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 469–473, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. "Attention is all you need." Advances in Neural Information Processing Systems 30, pages 5998–6008. Curran Associates, Inc., 2017.

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805, 2018.

[23] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." CoRR, abs/1803.01271, 2018.

[24] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. "Distributed representations of words and phrases and their compositionality." Advances in Neural Information Processing Systems 26, pages 3111–3119. Curran Associates, Inc., 2013.

[25] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. "Deep contextualized word representations." arXiv preprint arXiv:1802.05365, 2018.

[26] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. "Improving language understanding by generative pre-training." URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper.pdf, 2018.

[27] Zhiheng Huang, Wei Xu, and Kai Yu. "Bidirectional lstm-crf models for sequence tagging." arXiv preprint arXiv:1508.01991, 2015.

[28] Zeiler, Matthew D and Taylor, Graham W and Fergus, Rob and others. "Adaptive deconvolutional networks for mid and high level feature learning" IEEE International Conference on Computer Vision, vol 1, pages 2018-2025, ICCV 2011.

[29] Lee, Honglak and Grosse, Roger and Ranganath, Rajesh and Ng, Andrew Y. "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations" Proceedings of the 26th annual international conference on machine learning, pages 609–616, 2009 ACM.

[30] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. "Focal loss for dense object detection." CoRR, abs/1708.02002, 2017.

[31] Qiang Zhou. "Dialog Act Annotation for Chinese Daily Conversation." Journal of Chinese Information Processing, vol 31, pages 75–81, 2017.

[32] Lev Pevzner and Marti A. Hearst. "A critique and improvement of an evaluation metric for text segmentation." Comput. Linguist., 28(1):19–36, March 2002.