Two-Stage Probabilistic Approach to Text Segmentation

Yi-Chia Chen and Yi-Chung Lin

Abstract

For telephone-based spoken dialogue systems, the responses to users should be specific and short. Therefore, it is highly demanded to segment a topical text into specific event segments which can be used to answer users' queries. However, the lexical cohesion approach, which has been widely used to segment text into topics, is not suitable for segmenting text into smaller units, like events. In this paper, we present a two-stage approach to partition text into event segments. In the first stage, a trigram chunk tagger is used to label the segmentation tags. In the second stage, the unreliable segmentation tags are detected and then verified by a probabilistic verification model. Compared with the chunk tagger, the verification model can explore more contextual information and is less sensitive to the sparseness of training data. Experimental results show that the proposed two-stage approach significantly outperforms the chunk tagger approach. The improvements on precision and recall rates are 27% to 83% in different testing tasks.

1. Introduction

With the explosive growth of Internet information, people may easily get lost on the Internet while searching for the information they need. Therefore, there is an increasing need for providing users the information of what exactly they want. To meet this end, the online contents of different types, especially text, should be understood, to some extend, by machines. Current technology is not yet advanced enough to understand every detail of a lengthy text document. A good compromise given the state of technology is to divide text into segments of different topics and present only the segments related to topics of interested. Therefore, recently, the research on Topic Detection and Tracking (TDT [1]) has caught the attention of information industry.

The most fundamental issue of TDT is text segmentation. In the past, many methods are proposed to segment text into topics [2][3][4][5]. The common idea of these methods is using lexical cohesion features to detect the boundaries of a topic. Lexical cohesion features are mainly derived from the content words in a segment. Because text segments of the same topic tend to have similar vocabulary, the similarity between feature vectors can be used to identify whether consecutive segments belongs to the same topic.

The lexical cohesion approach is not suitable to partition a topic segment into smaller units, such as events. Since a topic consists of several directly related events [6], it usually includes several sentences up to several paragraphs. A text of such length is too long to read or to listen via telephone. Therefore, segmenting text into topics is not suitable for telephone-based applications. The unit of segmentation should be narrowed down to an event, which usually consists of one or several sentences describing something that happens at a specific time and place. Unlike the topic segment, an event segment is too short to contain many content words to provide reliable lexical cohesion information. Besides, since consecutive event segments of a topic are related, they tend to contain similar content words. Therefore, the lexical cohesion approach cannot well identify the boundary of different events in a topic.

In this paper, we propose a two-stage probabilistic approach to split text into event segments. In the first stage, the text segmentation problem is modeled as tagging problem. A trigram segmentation model is proposed to find the most probable segmentation tag sequence. In the second stage, the unreliable segmentation tags are first identified and then verified by a robust probabilistic model that is less sensitive to the sparseness of training data. The experimental results show that the proposed two-stage approach significantly outperforms the chunk tagger [7] approach. In the task of segmenting crisis warnings, the improvements on precision and recall rates are 83% and 70%. In the task of segmenting Switchboard [8] transcripts, the improvements on precision and recall rates are 59% and 27% respectively.

2. Segmentation by trigram chunk tagger

The process of text segmentation can be modeled as the tagging process (chunk tagger). Every word in the text is assigned a tag. If a word is the first word of a segment, it should be tagged as “1”. Otherwise, it should be tagged as “0”. The tagging process is formulated to find the most probable tag sequence \( \hat{T} \) for the given word sequence \( W \) as follows.

\[
\hat{T} = \arg \max_T P(T|W) = \arg \max_T \frac{P(T,W)}{P(W)}
\]

\[
= \arg \max_T P(T,W)
\]

(1)

where \( T \) denotes one possible tag sequence of \( W \). Let \( W \) is comprised of \( n \) words and \( w_i \) denote the \( i \)-th word. As a result, \( T \) is also comprised of \( n \) tags. Let \( t_i \) denote the \( i \)-th tag in \( T \) and \( \{t_i\} \) denotes the pair of \( w_i \) and \( t_i \). Then, the last probability term in the above equation is rewritten as:

\[
P(T,C) = P(\{t_i\}^n) = \prod_{i=1}^n P(\{t_i\} \mid t_{i-1}) = \prod_{i=1}^n P(t_i \mid \{\tau_{i-2}, \tau_{i-1}\}),
\]

(2)
where \( \tau_i^n \) is the shorthand notation for “\( \tau_1, \tau_2, \ldots, \tau_n \)”.

According to equations (1) and (2), we define the following scoring function to select the most probable tag sequence.

\[
S_{\text{trigram}}(\tau_i^n) = \sum_{i=1}^{n} \log P(\tau_i | \tau_{i-2}, \tau_{i-1})
\]

(3)

This scoring is intrinsically represents a trigram model, in which a word-tag pair depends on its previous two word-tag pairs.

### 3. Problem of sparse data

Theoretically, the number of the probability parameters of the trigram scoring function is the cubic power of the number of possible word-tag pairs. This number is usually much larger than the number of training data we can afford. To reduce the estimation error of these parameters, we adopt the widely used back-off smoothing method [9] to estimate the trigram parameters that do not occur in the training data.

Although the back-off smoothing method can reduce some estimation error of trigram probabilities, it still has its limits. If the training data is too sparse, many of the trigram probabilities will back-off to their corresponding unigram probabilities (i.e., prior probabilities). For example, in the trigram segmentation model, if \( w_{i-1} \) is unseen in the training data, the probability \( P(\tau_i | \tau_{i-2}, \tau_{i-1}) \) will be backed-off to \( P(\tau_i) \). In this case, the information of the word-tag pair \( \tau_{i-2} \) is abandoned even if the pair \( \tau_{i-2} \) occurs many times in the training data.

### 4. Two-stage approach for text segmentation

In order to make use the context information as much as possible, we propose a two-stage approach to segment text. Figure 1 shows the block diagram of this approach. In the first stage, the trigram segmentation model is used to label every word with a segmentation tag. Then, in the second stage, the unreliable segmentation tags are detected and verified according to contextual information. The following two sections give the details of the detection phase and the verification phase in the second stage.

#### 4.1. Detection unreliable tags

In the trigram segmentation model, the trigram probability \( P(\tau_i | \tau_{i-2}, \tau_{i-1}) \) is estimated according to the back-off smoothing method [9]. In brief, the back-off smoothing method adopts two different strategies to smooth \( P(\tau_i | \tau_{i-2}, \tau_{i-1}) \) if the trigram \( (\tau_{i-2}, \tau_{i-1}, \tau_i) \) is unseen in the training data. First, if the bigram \( (\tau_{i-2}, \tau_{i-1}) \) is observed in the training data, the Good-Turing’s estimate is used to assess the trigram probability. Second, if the bigram \( (\tau_{i-2}, \tau_{i-1}) \) is also unseen, the most distant word-tag pair (i.e., \( \tau_{i-2} \)) is ignored and the bigram probability \( P(\tau_i | \tau_{i-1}) \) is assigned to \( P(\tau_i | \tau_{i-2}, \tau_{i-1}) \). If \( \tau_{i-1} \) is also unseen, the prior probability \( P(\tau_i) \) is used instead.

Although the back-off method avoids assign zero probability to unseen trigram, we still concern the reliability of the probability. Therefore, if the trigram \( (\tau_{i-2}, \tau_{i-1}, \tau_i) \) is unseen, the segmentation tag \( \tau_i \) is considered as unreliable and need to be further verified.

#### 4.2. Verification according to contextual information

For an N-gram based model, the sparser training data it faces, the less contextual information it can make use of. Therefore, a more robust model that is not only benefited from contextual information but also less sensitive to the sparseness of training data is highly demanded. In our design, once a segmentation tag is determined to be unreliable, it will be verified according to its neighboring words and their hypothetical segmentation tags provided by the trigram segmentation model.

The verification process is actually a retagging process, in which more contextual information is explored to assign the segmentation tag again. Let \( \hat{\tau}_j \) indicate the segmentation tag that the trigram segmentation model selects for \( w_j \). Let \( j \) indicate the position index of one unreliable segmentation tag spotted in the detection phase. Then, the retagging process is formulated as follows.

\[
\hat{\tau}_j = \arg \max_{\tau_j} P(\tau_j | w_j^n, \hat{\tau}_j^{i-1}, \hat{\tau}_j^{n}),
\]

(4)

where \( w_j^n \) is the shorthand notation for “\( w_1, \ldots, w_n \)”, \( \hat{\tau}_j^{i-1} \) for “\( \tau_1, \ldots, \hat{\tau}_{j-1} \)” and \( \hat{\tau}_j^{n} \) for “\( \hat{\tau}_{j+1}, \ldots, \hat{\tau}_n \)”. The probability term in the above equation is further approximated as:

\[
P(\tau_j | w_j^n, \hat{\tau}_j^{i-1}, \hat{\tau}_j^{n}) \approx P(\tau_j | w_j^L, \hat{\tau}_j^{i-1}, \hat{\tau}_j^{R}),
\]

\[
P(w_j^L, \hat{\tau}_j^{i-1}, \hat{\tau}_j^{R} | \tau_j) P(\tau_j),
\]

(5)

\[
= \frac{P(w_j^L, \hat{\tau}_j^{i-1}, \hat{\tau}_j^{R} | \tau_j) P(\tau_j)}{P(w_j^L, \hat{\tau}_j^{i-1}, \hat{\tau}_j^{R})},
\]

where \( L \) and \( R \) are positive integers which define the window sizes for left context and right context respectively. Because
the prior probability \( P(w_{j-R}^j \mid \tilde{t}_R^j, \tilde{t}_{j+1}^j) \) is a constant, it can be ignored without changing the rank of all competing tags. The last conditional probability in equation (5) is further derived as:

\[
P(w_{j-R}^j \mid \tilde{t}_R^j, \tilde{t}_{j+1}^j, \tilde{t}_j^j) \approx \prod_{k=j}^{j+R} P(w_k \mid \tilde{t}_k^j) \times \prod_{k=j+1}^{j+R} P(\tilde{t}_k \mid \tilde{t}_j^j) = \prod_{d \in [L, R]} P(w_{j-d} \mid \tilde{t}_{j-d}) \times \prod_{d \in [L, R], d \neq 0} P(\tilde{t}_{j-d} \mid \tilde{t}_j^j)
\]

According to equations (5) and (6), equation (4) can be rewritten as:

\[
\hat{t}_j = \arg \max_{t_j} \log \left( \prod_{d \in [L, R]} P(w_{j-d} \mid \tilde{t}_{j-d}) \times \prod_{d \in [L, R], d \neq 0} P(\tilde{t}_{j-d} \mid \tilde{t}_j^j) \right)
\]

\[
= \arg \max_{t_j} \left\{ \sum_{d \in [L, R]} S_w_{j-d} \hat{t}_j + \sum_{d \in [L, R], d \neq 0} S_{\tilde{t}_{j-d}} \hat{t}_j \right\}
\]

where \( S_w_{j-d} \hat{t}_j = \log P(w_{j-d} \mid \tilde{t}_j^j) \), \( S_{\tilde{t}_{j-d}} \hat{t}_j = \log P(\tilde{t}_{j-d} \mid \tilde{t}_j^j) \) and \( S_{\tilde{t}_j} \hat{t}_j = \log P(t_j) \). In order to reduce the modeling error caused by the approximations made in formula derivations, the scores \( S_w_{j-d} \hat{t}_j \), \( S_{\tilde{t}_{j-d}} \hat{t}_j \) and \( S_{\tilde{t}_j} \hat{t}_j \) should be weighted. Therefore, we define the following scoring function to find the preferred segmentation tag:

\[
S_{\text{Verde}}(t_j) = \sum_{d \in [L, R]} a_d \cdot S_w_{j-d} \hat{t}_j + \sum_{d \in [L, R], d \neq 0} b_d \cdot S_{\tilde{t}_{j-d}} \hat{t}_j + b_0 \cdot S_{\tilde{t}_j} \hat{t}_j
\]

where \( a_d \), \( b_d \) and \( b_0 \) are the weighting factors. These weighting factors can be learned by automatic learning methods, such as adaptive learning [10].

5. Experiments and discussions

The proposed event segmentation model can also be applied to partition transcribed text into linguistic segments [12]. Therefore, the 1.2 million words of Switchboard transcripts are also used for experiments. This corpus includes 190,805 segments. 99% of this corpus is used to train probabilities and weighting factors. The remaining 1% of the corpus is used to assess the performances of different models.

Three different models are tested to show their capabilities of text segmentation. The first one is the trigram hidden segment model [12], named HS model. The second is the trigram chunk tagger [7], named CT model. The main difference between the chunk tagger and the hidden segment model is that the latter does not pair the word and the segmentation tag. The last one is the proposed two-stage model, named TS model, in which the results of the chunk tagger are verified according to the scoring function in equation (8) with \( L=2 \) and \( R=2 \).

Table 1 lists the performances of different models on segmenting weather crisis warnings. The chunk tagger (CT) outperforms the hidden segment model on recall rate. But their precision rates are almost the same. On the other hand, the two-stage model significantly outperforms the chunk tagger on both precision and recall rates. The precision rate is increased form 84.6% to 92.1%, which corresponds to 83% error reduction rate. The recall rate is also increased from 92.1% to 97.7%, which corresponds to 70% error reduction rate.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS model</td>
<td>85.0%</td>
<td>77.8%</td>
</tr>
<tr>
<td>CT model</td>
<td>84.6%</td>
<td>92.1%</td>
</tr>
<tr>
<td>TS model</td>
<td>97.4%</td>
<td>97.7%</td>
</tr>
</tbody>
</table>

The performances of different models on splitting Switchboard transcripts into linguistic segments are listed in Table 2. The performance rank changes. The chunk tagger becomes the worst one on both precision rate and recall rate. The reason will be discussed latter. However, the two-stage model is still the best one. Compared to the second best (i.e., the hidden segment model), the two-stage approach improves the precision rate from 56.5% to 73.9%, which corresponds to 40% error reduction rate. The recall rate is improved from 61.5% to 67.1%, which corresponds to 15% error reduction rate.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS model</td>
<td>56.5%</td>
<td>61.5%</td>
</tr>
<tr>
<td>CT model</td>
<td>36.2%</td>
<td>54.8%</td>
</tr>
<tr>
<td>TS model</td>
<td>73.9%</td>
<td>67.1%</td>
</tr>
</tbody>
</table>
As mentioned before, the chunk tagger outperforms the hidden segment model on segmenting crisis warnings. However, while segmenting Switchboard transcripts, the hidden segment model outperforms the chunk tagger. The rank changes because the levels of data sparseness in two cases are very different. If the training data is not sparse, the chunk tagger is better than the hidden segment model because it is more discriminative. Table 3 shows that, in the testing set of crisis warnings, only 4.5% of overall segmentation tags are determined to be unreliable. The reason is that the vocabulary of crisis warnings is very small. Only about 250 different words are used. Besides, the language style of crisis warnings is very regular. Therefore, only a small number of word-tag trigrams are unseen. However, on the contrary, the ratio of unreliable tags is quite large in the testing set of Switchboard transcripts. Near half of the segmentation tags are determined to be unreliable. The high level of data sparseness is due to the large vocabulary and complex language style.

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Ratio of unreliable tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crisis warnings</td>
<td>4.5%</td>
</tr>
<tr>
<td>Switchboard transcripts</td>
<td>47.3%</td>
</tr>
</tbody>
</table>

To sum up, no matter the training data is very sparse or not, the proposed two-stage model significantly outperforms the other two models. The superiority of the two-stage model indicates that the proposed detection and verification methods work effectively.

6. Conclusions

For telephone-based spoken dialogue systems, the responses to users should be specific and short. Therefore, it is highly demanded to segment a topical text into specific event segments which can be used to answer users’ queries. In the past, many lexical cohesion based methods have been proposed to segment text into topics. However, unlike topic segments, an event segment is too short to contain many content words to provide reliable lexical cohesion information. Besides, since consecutive event segments of a topic are related, they tend to contain similar content words. Therefore, the lexical cohesion approach cannot well identify the boundary of different events in a topic.

In this paper, we propose a two-stage probabilistic approach to segment text into events. In the first stage, the text segmentation problem is modeling as tagging problem. A trigram segmentation (chunk tagger) based model is proposed to find the most probable segmentation tag sequence. In the second stage, the unreliable segmentation tags are first identified and then verified by a robust probabilistic model that is less sensitive to the sparseness of training data. The experimental results show that proposed model significantly outperforms the chunk tagger model for the task of segmenting crisis warnings. The precision rate increases from 84.6% to 97.4%, which corresponds to an error reduction rate of 83%; the recall rate increases from 92.1% to 97.7%, which corresponds to an error reduction rate of 70%. For the task of segmenting Switchboard transcripts, the experimental results show that proposed model still significantly outperforms the chunk tagger model. The precision rate increases from 36.2% to 73.9% (i.e., 59% error reduction rate) while the recall rate increases from 54.8% to 67.1% (i.e., 27% error reduction rate). In both tasks, the proposed two-stage approach also significantly outperforms the hidden segment model. The improvements on precision and recall rates are 83% and 87% respectively in the task of segmenting crisis warnings. For the case of segmenting Switchboard transcripts, the improvements on precision and recall rates are 40% and 15% respectively.

7. Acknowledgments

This paper is a partial result of Project 3XS1B11 conducted by ITRI under sponsorship of the Ministry of Economic Affairs, R.O.C. Especially thanks to Ms. Yi-Fen Lo for her technical support on text segmentation.

8. References