Abstract—Punctuation restoration is important for Automatic Speech Recognition and the down-stream applications, e.g., speech translation. Despite the continuous progress on punctuation restoration, discriminating question marks and periods remains very hard. This difficulty can be largely attributed to the fact that interrogatives and narrative sentences are mostly characterized and distinguished by long-distance syntactic and semantic dependencies, which are not well modeled by existing models (e.g., RNN or n-gram). In this paper we propose to solve this problem by the self-attention mechanism of the Bert model. Our experiments demonstrated that compared the best baseline, the new approach improved the F1 score of question mark prediction from 30% to 90%.

I. Introduction

Naive automatic speech recognition (ASR) systems do not care about punctuation marks in the transcribed text. This is not a problem for applications with short utterances, e.g., voice-based inquiry or command. However, for applications with long utterances (e.g., conference transcribing and speech translation), punctuation is as important as lexical words. Punctuation restoration (PR) is the task of inserting appropriate punctuation marks in appropriate positions, which helps not only human readers but also down-stream NLP tasks (e.g., translation).

Two categories of information are often used for PR: acoustic information and textual information. Acoustic information involves prosody, pause duration between words, pitch-intensity, per-word timing [1], [2], [3], [4]. Textual information mainly focuses on contextual words and/or phrases. In this paper we focus on textual features only, as it is more related to the type of a punctuation rather than its position [5].

The central idea of text-based PR is to build a sequential labelling model that transcribes the input text sequence to an output punctuation marks (including an empty symbol). Most of the early studies of text-based PR employed statistical models such as conditional random fields (CRFs) [6], N-gram models [7], phrase-based MT models (PPMT) [8]. Recently, various neural networks have been utilized for PR, e.g., convolutional neural nets (CNN) [9], recurrent neural nets (RNN) [2], and bidirectional RNN (BRNN) [5], [10].

Despite the continuous progress on PR, it is found by several researchers that the question mark is very difficult to predict [4], [8], [10]. For example, Zelasko et al. [4] reported that about 20% of the question marks are mis-classified as periods, and Peitz et al. [8] reported a very low F1 score (27.5-33.2) for the question mark though the scores for other types of punctuation is rather high. We conjecture that this low F1 of the question mark prediction is that most of existing PR models are incapable of modeling the long-distance syntactic and semantic dependency in interrogatives sentences. For example, in the sentence “Are you planning to play the game in a bigger hall where there are many young children?”, the question mark is fully determined by the two words “Are you” at the beginning. Note that acoustic information is less useful to predict question marks, as interrogatives in English are not strongly associated with any prosodic patterns.

In this work, we use the deep bidirectional transformer (Bert) model [11] to tackle the log-distance dependency. Specifically, the self-attention mechanism of Bert allows it learning and utilizing long-distance syntactic and semantic dependencies, as the PR at a particular position will look up the entire sentence. Our experiments demonstrated that this new approach works surprisingly well for question mark prediction: it improved the F1 score from 30% to 90%.

II. Methods

A. Self-attention

Attention is a powerful mechanism in sequential modeling. This idea was firstly proposed by [12] to align the source and target sentences in machine translation, and then was applied to a broad range of sequence-to-sequence tasks [13], [14], [15]. In the net shell, the attention mechanism focuses on some particular locations of the source sequence when inferring the target sequence, and so the decoder knows which information to express at each inference step. The key ingredient here is that where to focus at each step is formulated as a flexible function (usually a neural net) that can be learned from data.

Although the attention mechanism was originally proposed for sequence alignment (mapping), it was recently extended by [16] to model individual sequences. The basic idea is to ‘enrich’ the semantic load of an element in a sequence by looking at its context that could be very long, tanks to the attention mechanism. This attention within a single sequence is called self-attention.

This work was supported by the National Natural Science Foundation of China No. 61633013. Dong Wang is the corresponding author.
Fig. 1. Self-attention Mechanism.

Fig. 1 illustrates the self-attention mechanism, where $E_t$ is the $t$-th element of the input sequence, $K_t, Q_t, V_t$ are the key, query and value derived from $E_t$, respectively. In order to represent $E_t$, all the elements $E_i$ are attended via $K_i$ by $E_t$ through $Q_t$. This is formulated as follows:

$$\alpha_{t,i} = \text{softmax}(Q_t^T K_i)V_i$$

$$C_t = \sum_i \alpha_{t,i}V_i,$$

where $C_t$ is the encoding of $E_t$. In Google’s paper [16], this is represented as a matrix form as follows:

$$C = \text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V,$$

where $d_k$ is the dimension of the keys.

The key advantage of self-attention is that the encoding $C_t$ encapsulates the information of all the sequence and so can capture long-distance dependency. This capability has been extensively used in many tasks [17], [18], [19], [20], [21].

B. Bert-based punctuation restoration

A very successful application of the self-attention mechanism is in the Bidirectional Encoder Representations from Transformers (Bert) model [11]. This model consists of a bunch of self-attention layers and full-connection layers, stacked alternatively. The self-attention layer consists of multiple heads [16], and is trained with a masked language modeling (MLM) task. This training can utilize a large amount of training data and learn very powerful word-level and sentence-level semantic representations of natural languages. It has been found that using the MLM pre-trained model can improve a multitude of downstream tasks in a significant way [11]. In this paper, we use Bert to perform punctuation restoration.

Fig. 2 shows the architecture of the Bert-based PR system. For each position that we want to examine if a punctuation should be inserted, a MASK token is inserted into the sequence. The masked-augmented sentence fragment is then projected into a sequence of continuous embeddings that consists of three parts: word embedding $E_w$, sentence embedding $E_s$ and position embedding $E_p$. The embedding sequence is then fed into the Bert model, and the output of the model is the probability $p(w)$ of all the words in the vocabulary, including the punctuation marks. If a punctuation mask $q$ achieves the highest probability, then $q$ will be inserted into the position of the Mask token.

To avoid false alarms, a simple rule was designed to filter out less confidence predictions:

- its probability $p(q)$ should be larger than a threshold $\theta$, i.e., $p(q) > \theta$;
- the margin between its probability $p(q)$ and the probabilities of all other tokens should be larger than a thread $\xi$, i.e., $(p(q) − p(w))/p(q) > \xi \forall w$.

Note that the punctuation prediction could be in an sequential style or individual style. In the sequential style, the prediction is sequentially and the punctuation marks that are predicted already will be used to predict for later positions; and in the individual style, all the positions are predicted independently. Our experiments showed that the individual style performs better, probably due to the error accumulation in the sequential style.

III. Experiments

A. Datasets

In this work we focus on English punctuation restoration, and choose to use three database to test our proposal: Europarl_v7, News Commentary, and TED. All these databases are in English, and have been widely used in former research. The details of these three databases are as follows:

- Europarl_v7: A corpus extracted from the proceedings of the European Parliament and prepared by Philipp Koehn [22]. It contains transcriptions of speeches by members of the parliament, and so it
is more of spoken style rather than writing style. Additionally, this corpus is rather formal as all the speeches are given officially. This corpus is divided into three parts: training, validation and testing, which consists of about 2 M sentences, 10 K sentences, 10 K sentences respectively.

- News Commentary: A corpus consists of political and economic commentary. This corpus was originally provided by the WMT challenge for training MT systems, and the source is taken from CASMACAT by J. Tiedemann [23].
- TED: A corpus extracted from the TED conferences. More than 3,400 talks’ transcriptions and translations are available on TED’s website\(^1\) created by volunteers.

### B. Pre-trained Bert

In the first experiment, we use the pre-trained Bert model released by Google\(^2\). We choose the Bert-Base model, which involves 12 layers, 768 hidden units per layer, and 12 heads in the self-attention layer. Our focus is the discrimination between periods and question marks. Three state-of-the-art approaches are chosen for a comparative study: BRNN[10], Hidden-N-gram[8], and PPMT[8]. Three metrics are used to evaluate the performance: precision, recall, and F1-score.

The results are summarized in Table I, where the results of the comparative systems are duplicated from the original papers. It can be observed that the Bert-base model works surprisingly well on predicting question marks and periods. The F1-score is improved from 89.8% to 94.8% for periods, and more significantly, the improvement for question marks is from 30% to 90%.

A more careful study reveals that the significant F1 improvement for question marks is mainly due to the increasing of recall, which is increased from 23% to 90%. This means that the main advantage of the Bert model is to retrieve the patterns of interrogatives, rather than discriminating patterns of different sentences. This is well understood: all the comparative models can learn local patterns only, so they can not discover long-distance dependency that is important for signifying interrogatives, leading to a large missing rate. In contrast, the Bert model, due the inner self-attention mechanism, can learn dependencies of any distance, hence is sensitive in detecting existence of question marks.

### C. Fine-tune and retrain

In the previous section, the pre-trained Bert model is used for PR directly. In this section, we examine how fine-tuning can adapt the model for the PR task. Additionally, we also re-train the model from scratch using the same data for fine-tuning. We also use training set of the Europarl\(_v7\) corpus to perform the fine-tuning and retraining, with the learning rate set to \(10^{-5}\) and \(10^{-4}\) respectively, and the batch size set to 25.

The training process is shown in Fig. 3, where the left picture shows the change of the loss function value during the fine-tuning/retraining process, on both the training and test sets; and the right picture shows the change of the masked LM accuracy. It can be observed that fine-tuning shows much better performance than re-training. This is expected as fine-tuning leverages the rich knowledge learned during the big-data pre-training.

Table II summarizes the results of the pre-trained Bert, the fine-tuned Bert and the re-trained Bert. All the results are reported on the Europarl\(_v7\) dataset. In order to have a more global picture of performance of different systems, three types of punctuations are reported: comma, question mark and period. It can be seen that for both comma and question mark, fine-tuning offers significant performance improvement: 8.5% for comma and 3.2% for question mark in terms of F1. For period, the fine-tuning does not show significant contribution. The significant improvement on comma can be attributed to the flexibility of this type of punctuation: different authors and different genres may exhibit significantly different behavior in using commas, so the adaptation by fine-tuning is effective. The same reason also explains why the performance of periods is not notably improved.

Finally, the retrained-model performs similar on periods, but better on commas and question marks. However, the improvement on commas and questions marks is not as significant as in the case of fine-tuning. This is consistent with the trend in Fig. 3 and demonstrates the effectiveness of large-data pre-training.

### D. Showcases

To observe how subtle changes in a sentence affect the prediction of punctuations, we present a few examples as shown in Fig. 4. In this case, the sentence is changed just a bit but the probabilities of different punctuation types predicted by Bert are significantly changed, even if the change is at the beginning of the sentence that is far from the position where the punctuation is predicted. This clearly demonstrated how the long-distance dependency between the indicative pattern at the beginning and the punctuation type at the end of the sentence has been learned by Bert.

We played this toy game with Chinese sentence as well. The difference between Chinese and English is that in Chinese, the punctuation type is not determined by clear syntactic patterns, but more semantic meaning and modal particles, e.g., ‘ma’, ‘ne’. Since these modal particles are strong indicators of the following punctuation marks, we simply remove them from the sentences. The Chinese Bert model pre-trained by Google was downloaded and used in the experiment.

The results are shown in Fig. 5. It can be seen that these four sentences look quite similar (only few words

\(^{1}\)http://www.ted.com/

\(^{2}\)http://github.com/google-research/bert
TABLE I
Performance of Bert-based PR and comparative methods.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>Period</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>BRNN[10]</td>
<td>Europarl_v7</td>
<td>77.2</td>
<td>39.5</td>
</tr>
<tr>
<td>Hidden-N-gram[8]</td>
<td>Europarl_v7+TED+News Commentary</td>
<td>88.9</td>
<td>90.7</td>
</tr>
<tr>
<td>PPMT[8]</td>
<td>Europarl_v7+TED+News Commentary</td>
<td>89.0</td>
<td>87.5</td>
</tr>
<tr>
<td>Bert</td>
<td>Europarl_v7</td>
<td>98.5</td>
<td>91.7</td>
</tr>
<tr>
<td></td>
<td>Europarl_v7+TED+News Commentary</td>
<td>98.4</td>
<td>86.2</td>
</tr>
</tbody>
</table>

**Fig. 3.** The change of loss (a) and MLM accuracy (b) during fine-tuning and retraining.

TABLE II
Bert-base, Bert-base Fine-tune and Bert-base Retrain

<table>
<thead>
<tr>
<th>Europarl_v7</th>
<th>Comma</th>
<th>Period</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>Bert</td>
<td></td>
<td>51.9</td>
<td>56.9</td>
</tr>
<tr>
<td>Bert + Fine-tune</td>
<td></td>
<td>55.3</td>
<td>72.4</td>
</tr>
<tr>
<td>Bert + Retrain</td>
<td></td>
<td>54.0</td>
<td>69.1</td>
</tr>
</tbody>
</table>

are different), but the probabilities of different punctuation marks predicted by Bert are clearly different. This demonstrated that Bert can not only learn long-distance syntax dependency, but also learn long-distance semantic dependency.

IV. Conclusion
This paper investigated a Bert-based punctuation restoration approach. Attributed to the capability of learning long-distance dependencies of the self-attention layers, Bert can be used to tackle the problem in question mark prediction, for which the most difficulty is in the syntax (English) and semantic (Chinese) long-distance existing in interrogatives sentences. Experiments on three datasets (Europarl_v7, News Commentary and TED) showed that the F1-score of the question mark prediction was improved from 30% to 90% by using Bert. These significant improvements are largely attributed to the increased recall, demonstrating the Bert can discover long-distance patterns that cannot be found by conventional methods, and use these patterns to identify interrogatives.

References


