Towards Better Subtitles: A Multilingual Approach for Punctuation Restoration of Speech Transcripts

Nuno Miguel Guerreiro[d] (miguelguerreironuno@gmail.com), Ricardo Rei[a] (ricardo.rei@unbabel.com), Fernando Batista[b,c] (fernando.batista@inesc-id.pt)

[a] Unbabel, Rua Castilho, 52, 1250-069 Lisbon, Portugal
[b] INESC-ID Lisboa, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
[c] Iscte - Instituto Universitário de Lisboa, Av. Forças Armadas, 1649-026 Lisboa, Portugal
[d] Instituto de Telecomunicações, Instituto Superior Técnico - Torre Norte - Piso 10, Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal

Corresponding Author:
Fernando Batista
Sala 234A, INESC-ID Lisboa, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
Tel: (+351) 21 3100 390
Email: fernando.batista@inesc-id.pt

Towards Better Subtitles: A Multilingual Approach for Punctuation Restoration of Speech Transcripts

Nuno Miguel Guerreiro[d], Ricardo Rei[a], Fernando Batista[b,c,*]

[a]Unbabel, Rua Castilho, 52, 1250-069 Lisbon, Portugal
[b]INESC-ID Lisboa, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
[c]Iscte - Instituto Universitário de Lisboa, Av. Forças Armadas, 1649-026 Lisboa, Portugal
[d]Instituto de Telecomunicações, Instituto Superior Técnico - Torre Norte - Piso 10, Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal

Abstract

This paper proposes a flexible approach for punctuation prediction that can be used to produce state-of-the-art results in a multilingual scenario. We have performed experiments using transcripts of TED Talks from the IWSLT 2017 and IWSLT 2011 evaluation campaigns. Our experiments show that the recognition errors of the ASR output degrade the performance of our models, in line with related literature. Our monolingual models perform consistently in Human-edited transcripts of German, Dutch, Portuguese and Romanian, suggesting that *commas* may be more difficult to predict than *periods*, using pre-trained contextual models. We have trained a single multilingual model that predicts punctuation in multiple languages that achieves results comparable with the ones achieved by monolingual models, revealing evidence of the potential of using a single multilingual model to solve the task for multiple languages. Then, we argue that usage of current punctuation systems in the literature are implicitly dependent on correct segmentation of ASR outputs for they rely on positional information to solve the punctuation task. This is too big of a requirement for use in a real life application. Through several experiments, we show that our method to train and test models is more robust to different segmentation. These contributions are of particular importance in our multilingual pipeline, since they avoid training a different model for each of the involved languages, and they guarantee that the model will be more robust to incorrect segmentation of the ASR outputs in comparison with other methods in the literature. To the best of our knowledge, we report the first experiments using a single multilingual model for punctuation restoration in multiple languages.

Keywords: Punctuation marks, Intelligent Subtitles, Pre-trained embeddings, Speech transcripts, Sentence boundaries, Multilingual embeddings

1. Introduction

Most of the existing ASR systems focus on minimizing the Word Error Rate (WER), making few attempts to detect the structural information that is available in spoken texts. The text produced by a standard speech recognition system often consists of raw single-case words, without punctuation marks. Such text is difficult to read and sometimes even hard to understand because of the missing information. Moreover, the missing information, specifically punctuation, sentence boundaries, and capitalization, is also important for many types of automatic downstream processing, such as parsing, information extraction, dialog act modeling, Named Entity Recognition (NER), summarization, and translation (Zechner, 2002; Huang and Zweig, 2002; Kim and Woodland, 2003; Ostendorf et al., 2005; Jones et al., 2005; Makhoul et al., 2005; Shriberg, 2005; Matusov et al., 2006; Cattoni et al., 2007; Ostendorf et al., 2008; Liao et al., 2020). Several studies have also shown that the punctuation marks, or at least sentence boundaries, are important for machine translation (Matusov et al., 2006; Cattoni et al., 2007; Peitz et al., 2011).

Spoken language is typically less organized than textual material, which makes it a challenge to bridge the gap between spoken and written material. Despite being originally used mostly for marking breaths, punctuation is nowadays used for marking structural units, thereby

* Corresponding author.
Email addresses: miguelguerreironuno@gmail.com (Nuno Miguel Guerreiro), ricardo.rei@unbabel.com (Ricardo Rei), fernando.batista@inesc-id.pt (Fernando Batista)

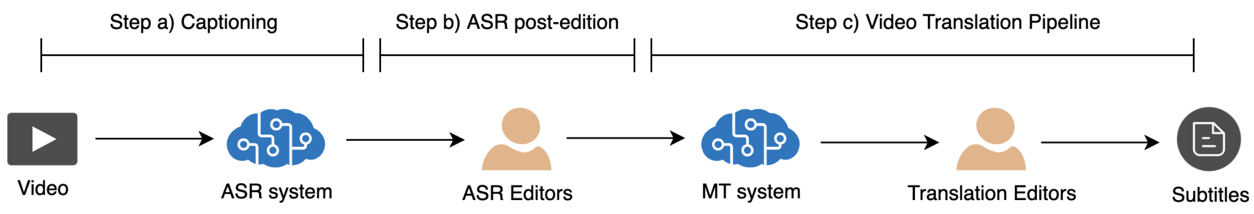

Figure 1: Unbabel Video Pipeline - Step a) consists of processing each video with an ASR system to create captions, in b) a human editor performs post-edition of the ASR texts and aligns the text with the audio. Finally, in step c), the captions are translated to other languages, first by using a customized MT system, and then by using humans to improve the resulting subtitles.

used to disambiguate meaning and to provide cues to coherence of the written text (Kowal and O'Connell, 2008). Inserting punctuation marks into spoken texts is a way of approximating such texts to written texts, keeping in mind that speech data is linguistically structured. Despite that, a punctuation mark may assume different behavior in speech, where the concept Sentence-like Unit (SU) is often used instead of sentence. Several punctuation marks can be considered for spoken texts, including: *comma*; *period* or *full stop*; *exclamation mark*; *question mark*; *colon*; *semicolon;* and *quotation marks*. However, most of these marks rarely occur and are quite difficult to insert or evaluate. Quotations and semicolons, for example, are often used inconsequently and in a highly variable way. Hence, most of the available studies focus on *full stop* and *comma*, which have higher corpus frequencies. A number of more restricted studies also consider the *question mark*.

This paper proposes a model for punctuation prediction based on pre-trained contextual embeddings. Our architecture is composed of three main building blocks: a pre-trained Transformer-based encoder model, an attention mechanism over the encoder layers, and a multinomial logistic regression classifier. We conduct evaluation experiments both in the transcripts of TED Talks from the IWSLT 2017 evaluation campaign, and in the English transcripts of TED Talks from IWSLT 2011, allowing to further compare our results with related work reported in the literature. We start by creating individual monolingual models, not only for English transcripts, but also for German, Dutch, Portuguese and Romanian transcripts. Then, we train a single and more flexible multilingual model for predicting punctuation across multiple languages, which is able to achieve fairly competitive results in a multi-language scenario, even surpassing the existing results for some of the languages. Using a single multilingual model to solve the task for multiple languages is of particular importance for our application, since training a different model for each language is a cumbersome and time-consuming process. The selected languages are based on the set of languages used in the scope of the IWSLT 2017 evaluation campaign, where the Italian language was replaced by Portuguese, a language that apart from being particularly relevant for our application, it is also the native language of the authors, a morphologically rich and low-resource language (in terms of available tools), and the sixth most widely spoken language worldwide. All the experiments reported in this work are based on lexical features only.

The rest of the paper is organized as follows: Section 2 introduces our video pipeline and presents our motivation. Section 3 presents an overview of the related literature. Section 4 describes our corpora, and the process of transferring the reference punctuation into to the ASR transcripts. Section 5 describes how the text data is transformed into labelled sequences and presents the building blocks of the model architecture. Section 6 describes the training setup, the evaluation metrics, and the conducted experiments. Section 7 reports the corresponding results achieved. Finally, Section 8 presents our most relevant conclusions, and pinpoints possible future directions.

2. System motivation: The Unbabel video pipeline

Unbabel is an AI augmented translation company that combines the speed and low cost of machine translation with a layer of human expertise powered by a community of human translators. Along with other translation services, Unbabel provides a very unique video subtitling solution that consists of a) processing each video with an ASR system adapted to the source language, b) manual post-edition of the ASR output by human editors, and c) perform the translation for other languages, first by using a customized machine translation system, and then by using humans to improve the resulting translations (Figure 1). Since the ASR transcripts, as they are generated, contain recognition errors, disfluencies, and missing information such as punctuation and sentence boundaries, step b) is the most time-consuming and expensive step throughout our video subtitling pipeline, especially, when the source language of the video is not English and the ASR system has worse quality. Hence, the enrichment of ASR text is of utmost importance to improve the edit-time and quality of the final subtitling pipeline especially for languages other than English.

3. Literature review

The sentence boundary detection problem is deeply connected to the punctuation recovery problem, especially when predicting punctuation like *full stops*, *question marks*, and *exclamation marks* (Shieber and Tao, 2003), which corresponds to sentence boundaries. These tasks provide a basis for further Natural Language Processing (NLP) tasks, and its impact on subsequent tasks has been analyzed in many speech processing studies (Harper et al., 2005; Mrozinsk et al., 2006; Ostendorf et al., 2008).

Recovering structural information in speech, and specially the detection of sentence boundaries, became the goal of an increasing number of studies in computational speech processing soon after developing the first Large Vocabulary Continuous Speech Recognition (LVCSR) systems. Early computational models for detecting punctuation marks and sentence boundaries typically involved a combination of n-gram language models and prosodic classifiers. The general Hidden Markov Model (HMM) framework, allowing the combination of lexical and prosodic cues, has been reported by a number of early studies on this task (Beeferman et al., 1998; Christensen et al., 2001; Kim and Woodland, 2001). A few years later, the detection of sentence boundaries was one of the main focus of the DARPA EARS rich transcription program (Liu et al., 2006). Discriminative models, such as Maximum Entropy (ME), and Conditional Random Field (CRF) have also been successfully used for this task (Huang and Zweig, 2002; Liu et al., 2006; Batista et al., 2007, 2008, 2009; Lu and Ng, 2010; Batista et al., 2010, 2012; Ueffing et al., 2013). Liu et al. (2005, 2006) and Batista et al. (2008) performed experiments comparing the predominant HMM approach with ME and CRF models, concluding that discriminative models generally outperform generative models.

Concerning machine translation, Stüker et al. (2006) describes the ISL machine translation system used in the TC-STAR 2006 evaluation. In this system, the output is enriched with punctuation marks by means of a case-sensitive, 4-gram language model and hard-coded rules based on pause duration. In this system, the punctuation is performed after the capitalization. Also concerning machine translation, Cattoni et al. (2007) report a system that recovers punctuation directly over confusion networks. This paper compares three different ways of inserting punctuation and concludes that the best results are achieved when the training corpus include punctuation marks in both languages, which means that the translation is performed from punctuated input to punctuated output. Peitz et al. (2011) compares different approaches for predicting punctuation in a speech translation setting, and show that punctuation prediction improves the quality of the final translation output.

Most of the recent approaches for punctuation restoration are based on embeddings and deep learning, either by solely using Recurrent Neural Networks (RNN), such as Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BLSTM), or by combining them with attention mechanisms. These approaches generally tackle the problem as a sequence-to-sequence or as a sequence labeling task (Tilk and Alumäe, 2015, 2016; Che et al., 2016; Klejch et al., 2017; Yi and Tao, 2019; Kim, 2019). Zelasko et al. (2018) use a BLSTM and a Convolutional Neural Network (CNN) to predict punctuation for conversational speech, using the Fisher corpus (Cieri et al., 2004). They use pre-trained GloVe embeddings, and conclude that embeddings play an important role in punctuation prediction. Nanchen and Garner (2019) combine several punctuation prediction models that use a range of features, from low level acoustics to high level semantics, and empirically show that features from different semantic levels are complementary.

The TED talks within IWSLT 2011 dataset (Federico et al., 2011) were used by a vast number of studies on punctuation prediction (Ueffing et al., 2013; Che et al., 2016; Yi et al., 2017; Kim, 2019; Yi and Tao, 2019; Yi et al., 2020). Ueffing et al. (2013) use Conditional Random Fields to combine a variety of different textual features to predict punctuation in spoken text. Che et al. (2016) proposes a purely lexical approach to predict punctuation in a 5-words window, using pre-trained word embeddings. Their experiments with the TED talks within IWSLT 2011 dataset surpass the results reported by Ueffing et al. (2013), when using lexical cues only. The authors combine all the speech segments into a single segment, taking into account that there will always be a punctuation mark at the end of each utterance. Yi et al. (2017) combines three models, a DNN model, a bidirectional RNN with attention mechanism, and a bidirectional LSTM with a CRF layer, for punctuation prediction. Their experimental results on the English IWSLT 2011 dataset achieved an overall 64.2% F-score on the reference data, outperforming previous state-of-the-art results. The authors conclude that the prediction of comma is more challenging when compared with the prediction of period and question mark in English. Kim (2019) proposes a deep recurrent neural network architecture with layer-wise multi-head attentions. Their experimental results on the IWSLT 2011 dataset achieve an overall 68.9% F-score, and 46.1% Slot Error Rate (SER) (Makhoul et al., 2005). Yi and Tao (2019) improves the approach described in Yi et al. (2017), by using self-attention and word and speech features, from the pre-trained Word2Vec and Speech2Vec embeddings. Their results on the English IWSLT 2011 dataset show that the self-attention model trained using word and speech embedding features outperforms the previous state-of-the-art, where word embeddings alone achieve an overall 69.4% F-score and the combination of word and speech features allow to reach 72.9% F-score on the reference data. Yi et al. (2020) outperform their previous results on the English IWSLT 2011 dataset, using a pre-trained BERT model, and only word embedding features, achieving an overall 77.8% F-score on the reference data.

Nowadays, transfer from pre-trained models such as BERT (Devlin et al., 2018) yields strong results on a wide variety of token-level tasks, both in monolingual and multilingual scenarios (Pires et al., 2019). These models consist of Transformer encoders which attend to bidirectional contexts during pre-training with a Masked Language Modelling (MLM) objective. MLM consists of corrupting the input sequence by randomly masking some tokens and asking the model to restore the original text in those positions. As a result of this training objective these models learn to represent words in a highly contextualized feature space that is informative for downstream tasks. Recently, RoBERTa (Liu et al., 2019) an iteration study of BERT, led to improvements over the previous model by: (i) training the model longer, on longer sequences, on larger batches and more data; (ii) removing the next-sentence prediction task, and; (iii) dynamically change the masked positions for a given sample during pre-training. XLM-R trains a masked language model on one hundred languages, using more than two terabytes of filtered CommonCrawl data, leading to significant performance gains for a wide range of cross-lingual transfer tasks. BERT and RoBERTa were recently used by Liao et al. (2020) to successfully perform a ASR post-processing task, aiming a producing readable texts from ASR transcripts.

Recent work in punctuation recovery using BERT have been reported by Cai and Wang (2019); Makhija et al. (2019). In both works, the problem of punctuation restoration is treated as a sequence labeling problem where BERT is used to encode the input sequence. In Cai and Wang (2019) the authors, for each position, sequentially add a mask token and try to decode a punctuation mark. As such for an input sequence with n tokens the model as to encode $n+1$ tokens n times. In Cai and Wang (2019) a more traditional approach is used, namely a BLSTM with a CRF. As such, BERT is only used as embeddings layer to extract word-level features that will be passed to the top models. We argue that pre-trained models such as BERT are powerful enough to solve the punctuation task with only a simple regression on top.

Concerning the punctuation of video video subtitles, Batista et al. (2007, 2010) describe a punctuation module for automatic subtitles of Portuguese Broadcast News. Their approach uses lexical, acoustic and prosodic information, and their results show that prosody is relevant for the detection of full-stops, commas and question marks. Klejch et al. (2016) presents a most recent study that tests three punctuation models using lexical features, and one that uses acoustic features. Their experiments using the English Language MGB Challenge data Bell et al. (2015), consisting of about 1.600 hours of BBC TV recordings, show that a lexical-based neural machine translation performs significantly better than the other systems. Klejch et al. (2017) extends this previous work, and concludes that combining acoustic features with lexical features leads to a significant improvement over lexical features only. Ákos Tündik et al. (2020) propose a RNN-based model suitable for on-the-fly close captioning of Hungarian and English broadcast data. Their subjective tests confirmed that punctuation significantly contributes to the quality of the subtitles.

4. Corpora

Our models were trained on data from the IWSLT 2017 evaluation campaign[1], consisting of human-edited transcripts of TED Talks. We used the English (EN), German (DE), Dutch (NL), Portuguese (PT) and Romanian (RO) subsets to train different monolingual models for each of these languages. The training sizes for EN, DE, NL, PT and RO consist of 1705, 1399, 1902, 1474 and 1812 different TED talks, respectively. The development and test sets for each of the languages consist of 11 and 8 talks, respectively.

[1] `http://workshop2017.iwslt.org/59.php`

Moreover, we have defined a training, development and test multilingual corpus by concatenating the training, development and test sets for each one of the languages.

In order to compare our results in English with results in the literature, we also use the test data of IWSLT 2011 (Federico et al., 2011), both the automatic transcripts of English TED Talks, produced by 4 different ASR systems, and the corresponding human-edited transcripts. In particular, we constructed the test set for the ASR track by concatenating the outputs of all 4 different systems. Both the human-edited and ASR transcripts are organized into 818 segments of words, each of which ends either with a *full stop* or a *question mark*. At the moment, we do not have information on how these segments were produced: they may have been manually created, automatically produced based on silence thresholds, or produced by an ASR system.

We also have transferred the punctuation marks from the reference (human-edited) transcripts into the ASR transcripts, in order to be able to perform the evaluation on the ASR transcripts. The two datasets were aligned using a variant of Minimum Edit Distance, adapted for punctuation marks. The ASR transcripts contain errors that prevent a perfect alignment, and may pose problems to the location of the candidate punctuation mark on the ASR data. Therefore, we have decided to transfer a punctuation mark, if **at least** the left or the right context word have been recognized correctly, including the beginning or the end of a segment. Figure 2 presents examples of the alignment results (RES), taking into account the reference (REF) and the ASR data. The second punctuation mark in the second example is an example of a punctuation mark that was not transferred because both left and right context words differ.

Our approach for aligning the data is different from the one described in Ueffing et al. (2013), and probably adopted by subsequent studies (Che et al., 2016; Yi et al., 2017; Yi and Tao, 2019; Yi et al., 2020), that restrict the evaluation to those punctuation marks whose context words to the left and the right have been recognized correctly. Such restriction prevents the authors from assigning punctuation marks to locations containing recognition errors, which leads to better punctuation performance, since recognition errors are known to have a significant impact on the punctuation performance (Liu et al., 2006; Batista et al., 2012; Tilk and Alumäe, 2015, 2016; Che et al., 2016; Yi et al., 2017; Ákos Tündik et al., 2020). For example, such an approach would prevent from transferring most of the punctuation marks in the examples of Figure 2.

The reference transcripts may contain abbreviations

REF: <And that’s my cousin and my sister’s dog , Gabby .>
ASR: <and that’s my cousin and my sister’s dog every>
RES: <And that’s my cousin and my sister’s dog , every .>

REF: <As Juan said , it’s the condition that scientists call synesthesia , an unusual cross-talk between the senses .>
ASR: <as kwan said it’s a condition that scientists call soonest easier when usual course talk between the fences>
RES: <As kwan said , it’s a condition that scientists call soonest easier when usual course talk between the fences .>

REF: <Well you can’t get much bigger than Pi , the mathematical constant .>
ASR: <well you can’t get much bigger than empowering the mathematical constant>
RES: <Well you can’t get much bigger than empowering , the mathematical constant .>

REF: <is it a happy word , or a sad word ?>
ASR: <is it a happy word was sad word>
RES: <is it a happy word , was sad word ?>

REF: <Well if 100 , if we think of 100 as being like a square , 75 would look like this .>
ASR: <well it’s a hundred we think of a hundred as of being a like a square seventy five would look like this>
RES: <Well it’s a hundred , we think of a hundred as of being a like a square , seventy five would look like this .>

Figure 2: Examples of alignments between the reference and the ASR transcripts, and the resulting punctuation of the ASR transcripts.

and numeric values, while the ASR transcripts do not include abbreviations, and numbers are usually written as text. Therefore, in order to be performed correctly, the alignment process must also include a normalization step. In order to account for these differences, we have manually corrected the automatic alignment results. The last example of Figure 2 illustrates that situation, where the number *100* is aligned with the word *hundred*. All the corpora described above, including the manually corrected references, are publicly available for download[2].

5. Model Overview

We treat the problem of punctuation restoration as a sequence labeling problem such that we want to classify

[2] `https://unbabel-experimental-data-sets.s3-eu-west-1.amazonaws.com/video-pt2020/IWSLT-punkt.tar.gz`

whether a word in the sequence is followed by a punctuation mark. In this section, we start by describing how we transform the text data into labelled sequences, and then by presenting the building blocks of the model architecture.

5.1. Data Preprocessing

We define two different sets of classes: a set with 3 labels (O, C, P) and a set with 4 labels (O, C, P, Q). Following the punctuation settings of Che et al. (2016) and Tilk and Alumäe (2016), O means no punctuation mark followed; periods, exclamation marks or semicolons are classified as P, while commas, colons or dashes are classified as C. Question marks are classified as P for the 3-labels set and as Q for the 4-labels set.

It is also central that the model does not rely on positional information to solve the task. As such, we do not input a single segment sequence but a chunk of segments up to 300 tokens. This value was set taking into consideration the maximum sequence length that our encoder model can process plus a safety margin to take into account sub-word token splits. As Che et al. (2016) reports, the performance of punctuation prediction is largely influenced by the average number of punctuation marks per sequence in the corpus. Notice that it is expectable that there will always be a period or a question mark at the end of a single segment. Thus, if we were to train the models at the segment-level, and if we took into consideration these very same punctuation marks at the end of each segment during evaluation, the metrics for the PERIOD/QUESTION labels would likely be overestimated. The results that will be presented in Section 7 further support this notion.

To create the source sentence, we strip all punctuation and capitalization from the target sentence. An example of this procedure can be seen in Table 5.1.

Target		He is, according to the critics, an amazing player. What do you think?
Source		he is according to the critics an amazing player what do you think
Labels	*3-labels*	O C O O O C O O P O O O P
	4-labels	O C O O O C O O P O O O Q

Table 1: Source sentence and target labels construction – for the two sets of labels – for a given sentence. Note that apart from removing punctuation from the target, lowercasing was also applied to create the source sentence, and the source sequence consists of two different sentences. Here *O* stands for no punctuation, and *C*, *P* and *Q* are respective to the COMMA, PERIOD and QUESTION labels, respectively.

5.2. Model Architecture

Figure 3 presents our proposed model architecture, composed by three main building blocks: a Transformer-based encoder model (Vaswani et al., 2017), an attention mechanism over the encoder layers, and a multinomial logistic regression classifier.

5.2.1. Encoder Model

The main building block of the proposed architecture, the encoder model, consists of either RoBERTa or XLM-RoBERTa. While the latter can be used on a multilingual setting – it has been trained on 100 languages – the former can only be used on a monolingual English setting.

Both versions of RoBERTa and XLM-RoBERTa used in this work consist of 12 Transformer encoder layers with hidden size of 768. RoBERTa divides texts into bytes and uses Byte-Pair Encoding (BPE) (Sennrich et al., 2016) to build up its 50k–token vocabulary, whereas XLM-R increases the vocabulary size to 250k tokens in order encompass different languages across different scripts.

5.2.2. Layer Attention

Given an input sequence $\boldsymbol{x} = [x_0, x_1, \ldots, x_n]$, the encoder will produce an embedding $\boldsymbol{e}_{x_j}^{(\ell)}$ for each token x_j and each layer $\ell \in \{0, 1, ..., 12\}$.

The final hidden state, $\boldsymbol{e}_{x_j}^{(12)}$, of each token is commonly used for token-level tasks. However, as reported in Tenney et al. (2019), the BERT model captures, within the network, linguistic information that is relevant for downstream tasks. Thus, in this work, we used the approach in (Peters et al., 2018; Kondratyuk and Straka, 2019) to encapsulate information from all encoder layers into a single embedding, $\boldsymbol{e}_{x_j}$, for each token by using a layer-wise attention mechanism.

This embedding will then be computed as:

$$\boldsymbol{e}_{x_j} = \gamma \boldsymbol{E}_{x_j}^{\top} \boldsymbol{\Lambda} \tag{1}$$

where γ is a trainable scaling factor, $\boldsymbol{E}_{x_j} = [e_{x_j}^{(0)}, e_{x_j}^{(1)}, \ldots e_{x_j}^{(12)}]$ corresponds to the vector of layer embeddings for token x_j, and $\boldsymbol{\Lambda} = \text{softmax}([\lambda^{(1)}, \lambda^{(2)}, \ldots, \lambda^{(12)}])$ is a vector constituted by the layer scalar trainable parameters which are shared for every token. Intuitively, higher $\lambda^{(\ell)}$ values will be assigned to layers that hold more relevant information to solve the task. In order to redistribute the importance through all the model layers and avoid overfitting of the model to the information contained in any single layer, we used layer dropout, devised by Kondratyuk and Straka (2019), in which each the weight $\lambda^{(\ell)}$ is set to $-\infty$ with probability 0.1.

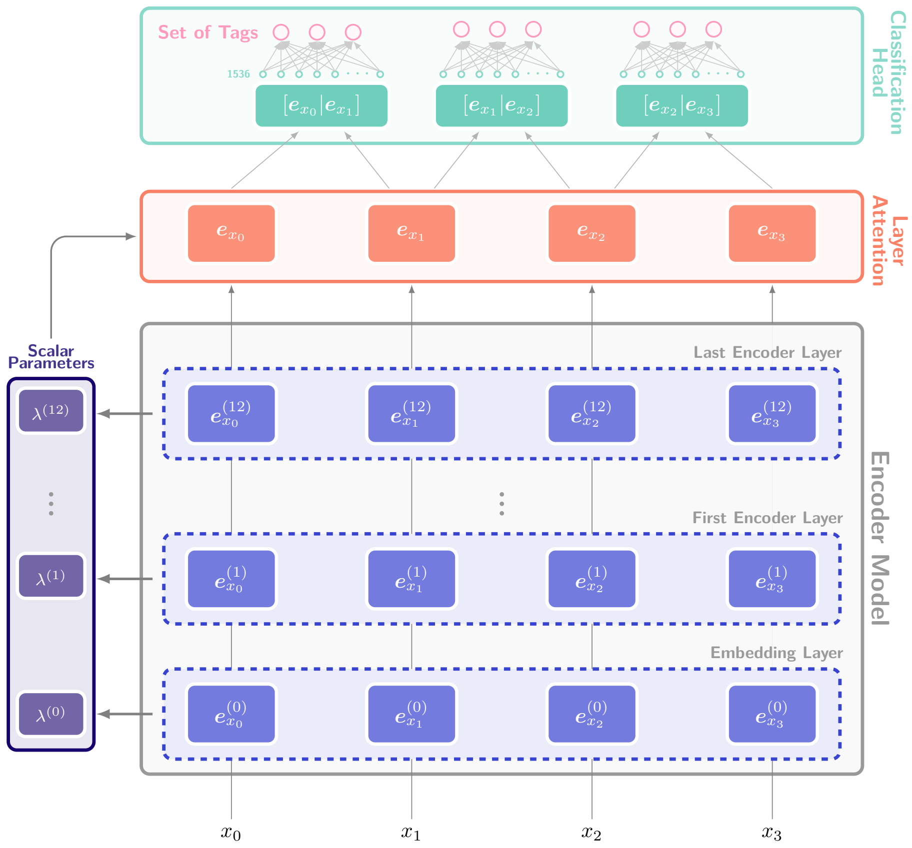

Figure 3: The architecture of our solution. The output of the encoder model passes through the Layer Attention (the scalar weights respective to each layer are trained simultaneously with the rest of the model) block respective to Equation (1). The representations of consecutive tokens are then concatenated and fed into a Multinomial Logistic Regression classifier that will predict whether a word should be followed or not by a punctuation mark. In this picture, the set of labels corresponds to the 3-labels set (O,P,C).

5.2.3. Classification Head

The final block of our architecture is a classification head that consists of a Multinomial Logistic Regression. As we want to classify whether a gap between two words should be filled with a punctuation mark, we start by concatenating the embeddings respective to consecutive tokens in the input sequence and then feed this joint representation to a multinomial logistic regression classifier that will output the predicted label.

6. Experiments

We apply our architecture on two different settings: a monolingual setting and a multilingual one. In this section, we will introduce the training setup and the evaluation metrics for our experiments.

6.1. Training Setup

We start training by loading the pre-trained encoder model and initializing both the layer attention and the linear projection in the classification head. Following the approach in (Howard and Ruder, 2018), we used gradual unfreezing and discriminative fine-tuning by dividing the set of parameters into two groups: the encoder parameters, that correspond to the parameters in the encoder model and the layer attention; and the classification parameters, which are respective to the parameters in the classification head. The encoder parameters are frozen during the first epoch and subsequently trained with a learning rate of 3×10^{-5}. The classification parameters are trained with a constant 5×10^{-5} learning rate and dropout probability of 0.1. Evaluation is performed at the end of each epoch and training stops when the validation SER does not improve for 3 consecutive epochs. At test time, the model with the best validation SER is selected.

6.2. *Evaluation Metrics*

All the evaluation presented in this paper uses the performance metrics: F1-score and SER (Makhoul et al., 2005). Only punctuated gaps between two words are considered as slots and used by these metrics. Hence, the punctuation SER is computed by dividing the number of punctuation errors (misses and false alarms) by the number of punctuated gaps in the reference.

6.3. *Conducted Experiments*

We start by showing the results for the models trained on the monolingual datasets and multilingual datasets. After that, we make some remarks regarding the training samples preparation, either for human-edited transcripts or ASR transcripts, and how it affects the results. By doing so, we establish a comprehensive comparison between our results and the results in the literature. Then, we analyze how a single multilingual model compares with the monolingual models to investigate whether the former is competitive with the latter ones and if insights gained through training with one language are transferred to another language.

7. Results

This section reports the results and provides an analysis of the experiments mentioned in Section 6.3.

7.1. *Monolingual Models*

We start by analyzing the results for models trained on a single language. It is important to recall that, due to the pre-training of the encoder models, RoBERTa can only be used as the encoder model of our architecture to solve the task for English whereas XLM-R can be used to solve the task for any of the 100 languages that it supports. Additionally, as most studies in the literature regarding punctuation restoration are based solely on solving the problem in English, we separate this section into two parts: one part that is entirely dedicated to English, and another that is dedicated to the other languages (German, Dutch, Portuguese and Romanian).

7.1.1. *English Language*

We report the results in Table 2. We can observe that RoBERTa slightly outperforms XLM-R as the encoder model of our architecture in all the tests. That is predictable, since RoBERTa was pre-trained only on English data and XLM-R was pre-trained with multilingual data. It is clear that the performance of our models is degraded for the ASR track. We argue that this is possibly due to the recognition errors of the ASR outputs and subsequent punctuation transfer (see Figure 2). We also note that as the proportion of QUESTION labels in the datasets is very small comparing with the proportion of PERIOD and COMMA labels, it was expected that the results when testing the models that were trained on 3 labels and those that were trained on 4 labels would be very similar. Finally, for the human-edited transcripts, we observe that the models are able to consistently classify the QUESTION label better than the PERIOD tag. This is not the case for the ASR transcripts. This can also be due to the alignment/recognition errors in the ASR – as questions in English are usually formulated with an auxiliary verb and subject inversion (e.g. Is it...?, Do you...?), if an error occurs in the alignment/recognition and those words do not feature in the reference, they will not act as a cue to predict the question mark, and the model, consequently, will not be able to predict it correctly. Figure 4 shows one example of such event. Nevertheless, the most frequent errors are due to situations where commas and periods can be used interchangeably. As our models do not rely in positional information to predict PERIOD and COMMA labels, the most frequent errors are predicting a COMMA for a given gap whose target reference is PERIOD, and vice-versa. Figure 5 depicts such a case. This issue will be discussed later on in Section 7.3.

Human-Edited Transcripts	
REF:	(...) is it a happy word, or a sad word ? (...)
PRED:	(...) is it a happy word or a sad word ? (...)
Aligned ASR Transcripts	
REF:	(...) he said that happy word , was sad work ? (...)
PRED:	(...) he said that happy word was sad. work (...)

Figure 4: Example of an excerpt of a test sample in which the ASR recognition errors led to a prediction error. Note that, for this particular example, the SER for the reference sample is 2/3, and for the ASR sample is 1/4.

7.1.2. *Other Languages*

We trained monolingual models using the 3 labels setting and XLM-R as the encoder model for the other languages considered in this study. Table 4 reports the corresponding results. We observe that the results are consistent across all languages. It is important to remark that comparisons across all languages are not exactly fair, since each language has its own set of features. Moreover, the transcripts for each of the talks in the datasets are obtained via translation, and these are not necessarily created by the same translators nor re-

Test	Track	Encoder Model	3 labels				4 labels				
			F1 Score			SER	F1 Score				SER
			PERIOD	*COMMA*	*Overall*		*PERIOD*	*QUESTION*	*COMMA*	*Overall*	
IWSLT 2011	Human Edited	RoBERTa	83.0	73.3	78.1	30.0	82.2	83.2	73.5	79.6	29.7
		XLM-R	81.1	72.2	76.7	31.9	80.9	76.5	72.3	76.6	32.5
	ASR	RoBERTa	77.3	60.5	68.9	53.5	76.6	64.8	60.4	67.3	54.2
		XLM-R	74.8	58.7	66.8	55.3	74.9	61.5	58.9	65.1	56.8
IWSLT 2017	Human Edited	RoBERTa	84.5	75.2	79.8	28.7	84.0	86.1	75.5	81.7	28.5
		XLM-R	81.9	73.4	77.7	31.3	81.9	82.9	73.5	79.4	31.5

Table 2: Metrics, in percentage, on the English test sets of IWSLT 2011 (Human Edited and ASR tracks), and IWSLT 2017 (Human Edited track) for the four different trained models (2 encoder models: XLM-R or RoBERTa; and 2 sets of labels: 3 labels or 4 labels setting).

Fail to predict PERIOD by predicting COMMA

REF: (...) today the average American spends about a week a year stuck in traffic jams . and that's a huge waste of time and resources . (...)

PRED: (...) today the average American spends about a week a year stuck in traffic jams , and that's a huge waste of time and resources . (...)

Fail to predict COMMA by predicting PERIOD

REF: (...) i'm going to do an international biennial , i need artists from all around the world . (...)

PRED: (...) i'm going to do an international biennial . i need artists from all around the world . (...)

Figure 5: Example of excerpts of two test samples in which the model fails in its predictions: in the first excerpt, it introduces a comma in a gap whose reference label was a period; in the second excerpt, it introduces a period in a gap whose reference label was a comma.

EN	Several years ago here at TED , Peter Skillman introduced a design challenge called the marshmallow challenge . And the idea's pretty simple : Teams of four have to build the tallest free-standing structure out of 20 sticks of spaghetti , one yard of tape , one yard of string and a marshmallow .
PT	Há uns anos , aqui no TED , Peter Skillman apresentou um desafio de projecto chamado "O desafio do marshmallow" . A ideia é muito simples . Equipas de quatro têm que construir a estrutura mais alta e estável , a partir de 20 barras de esparguete , um metro de fita cola , um metro de cordel e um "marshmallow" .

Table 3: Human edited transcriptions of an excerpt of a TED Talk in English and Portuguese. The punctuation for the two languages is introduced differently, leading to variance in the results across languages. Blue is used to represent inserted punctuation, green is used when the punctuation across the two languages is aligned, and red is used to represent a gap where a different punctuation mark was used across the two languages.

viewed by the same reviewers. In Table 3, we observe that, comparing a parallel excerpt transcript in English and Portuguese, the way the punctuation is introduced is different. Specifically, the transcript in Portuguese contains two more commas and one more period. Note that the period is introduced in a gap for which a colon was assigned in English. Actually, a colon was introduced in this very same gap in Dutch and English, whereas the others introduced a period. This reveals some inconsistency between translators on what is the correct punctuation mark to use in that case.

7.2. A Single Model for Multiple Languages

To the best of our knowledge, there is not a study in the literature for punctuation restoration for multiple languages making use of a **single** model. We tackle that by making use of the multilingual dataset, previously mentioned in Section 4, to train our model with XLM-R as the encoder model. Note that this pre-trained encoder model already encapsulates information for more than 100 languages. Thus, by training our architecture, we will end up finetuning the encoder's weights to solve the punctuation restoration task for five different languages. The right side of Table 4 shows the results of the single multilingual model for each one of the language's human-edited transcripts in the IWSLT 2017 test sets. The result for English is respective to the monolingual trained with XLM-RoBERTa as the encoder model. We can see that the single multilingual is very competitive with the monolingual models, even outperforming some of them in what comes to SER. Training a single model with data from five different languages had a bigger impact on the PERIOD label, for which the results are consistently better than those respective to the monolingual models. This might be due to a structural similarity in the way this type of punctuation is introduced across the different languages. From a semantic standpoint, it is expected that the COMMA label requires more understanding of the language when compared to the PERIOD label. These results evidence the potential of

IWSLT 2017 Test Set: Human-Edited Transcripts								
Language	Monolingual Models				Single Multilingual Model			
	F1 Score			SER	F1 Score			SER
	PERIOD	*COMMA*	*Overall*		*PERIOD*	*COMMA*	*Overall*	
EN	81.9	**73.4**	77.7	**31.3**	**83.0**	73.0	**78.0**	31.4
DE	86.3	**82.7**	**84.5**	**22.8**	**86.7**	77.7	82.2	27.1
NL	84.8	68.2	**76.5**	32.5	**84.9**	**69.2**	**77.1**	**32.0**
PT	83.5	69.0	76.2	36.0	**84.0**	**69.4**	**76.6**	**34.2**
RO	83.1	69.2	76.0	35.6	**83.7**	**70.8**	**77.2**	**34.2**

Table 4: Metrics, in percentage, on the human-edited transcripts test set of IWSLT 2017 respective to each of the five different monolingual models and the single multilingual model. All the models were trained with XLM-R as the encoder model.

IWSLT 2011 Test Set: Human-Edited and ASR Transcripts					
Track	F1 Score				SER
	PERIOD	*QUESTION*	*COMMA*	*Overall*	
Human Edited	82.6	78.8	73.1	78.2	31.2
ASR	76.4	63.0	59.9	66.4	54.2

Table 5: F1-Score, in percentage, on the human-edited and ASR transcripts test set of IWSLT 2011 respective to the single multilingual model.

using a single multilingual model to solve the task for multiple languages. This is of particular importance for our application, since training a different model for each language is a cumbersome and time-consuming process.

We also report in Table 5 results from evaluation of the single multilingual model in the IWSLT 2011 test sets for both human-edited and ASR tracks. By examining the results and comparing with those in Table 2, we can see that the single multilingual model performance is competitive with that of RoBERTa in both tracks, even being able to match it on the ASR track in what comes to SER.

7.3. Comparison with the Literature

In this section, we will compare our methods and results with those reported in Yi et al. (2020) as it stands as the current state of the art for this task. There are two main differences between our studies:

- we train and evaluate our models on chunks of segments (*chunk-level*), i.e., a training/testing sample consists of multiple segments, whereas Yi et al. (2020) train and test their models considering each segment as a training/test sample (*segment-level*);
- we consider alignment and recognition errors in our evaluation of the ASR outputs, whereas Yi et al. (2020), following the procedure in Ueffing et al. (2013), restrict the evaluation to those punctuation marks whose context words to the left and the right have been recognized correctly.

The models in Yi et al. (2020) were tested in the IWSLT 2011 test set, both for the human-edited transcripts and the ASR outputs. We will be using the 4 tags setting in our comparisons, as that is the setting that is used in their work. The model developed in that study is a monolingual English language, so we will be using our best monolingual English language model to draw the comparisons, i.e, the one with RoBERTa as the encoder model.

7.3.1. Training on Chunks of Segments

We observed that by training on chunks of segments, the model would not be able to entirely rely on positional information to solve the punctuation task. This is not the case for the models that are trained at the segment-level. The latter models develop an heuristic to assign the correct sequence of labels: i) label the last gap of a sample with a PERIOD label, and ii) do not label any gap that is not the last with a PERIOD/QUESTION label. This is mainly due to the fact that all the training samples for a sentence-level training contain a PERIOD/QUESTION in the last gap and, as they consist of a single sentence, it is highly likely that there are no other PERIOD/QUESTION in the segment. By training and evaluating at the segment-level, the PERIOD/QUESTION are easily predicted and there will be little to no conflict between the PERIODs and COMMAs. That is not the case for training and evaluation on chunks of segments, since the model cannot rely on heuristics to solve the task. Note that as the segments are concatenated, there will be multiple PERIODs/QUESTIONs and COMMAs in the sample.

To draw qualitative comparisons, we conducted three additional experiments: i) train a model with RoBERTa as the encoder model at the segment-level, ii) evaluate

IWSLT 2011 Test Set: Human-Edited and ASR Transcripts					
Track	**Model**	**F1 Score**			
		PERIOD	*QUESTION*	*COMMA*	*Overall*
Human Edited	Yi et al. (2020) best model	84.1	75.8	73.6	77.8
	Our best model	**98.1**	**88.2**	**84.0**	**90.0**
ASR	Yi et al. (2020) best model	79.5	69.6	**70.8**	73.3
	Our best model	**97.8**	**75.7**	67.8	**80.4**

Table 6: F1-Score, in percentage, on the human-edited transcripts test set of IWSLT 2011 respective to the best model of Yi et al. (2020) and our best model trained at the segment-level. The test set is such that each sample corresponds to a single segment.

IWSLT 2011 Test Set: Human-Edited Transcripts					
Train Level	**Test Level**	**F1 Score**			
		PERIOD	*QUESTION*	*COMMA*	*Overall*
segment-level	segment-level	**98.1**	**88.2**	**84.0**	**90.0**
	chunk-level	14.0	4.17	57.9	25.3
chunk-level	segment-level	**89.3**	78.4	**79.7**	**82.5**
	chunk-level	82.2	**83.2**	73.5	79.6

Table 7: F1-Score, in percentage, on the human-edited transcripts and ASR test sets of IWSLT 2011 respective to the models trained at the segment-level and at the chunk-level. Evaluation was also performed at the sentence-level and at the chunk-level.

the model trained at the chunk-level in the IWSLT 2017 test set at the segment-level, i.e., each test sample corresponds to a single segment, and iii) evaluate the model trained at the segment-level in the IWSLT 2017 test set at the chunk-level, i.e, each test sample corresponds to chunks of segments.

To train the model at the segment-level, we used the same training setting described in Section 6.1. The only difference resides in the data itself. In Table 6, we report the results of our best model **trained and evaluated at the segment-level** and the best model reported in Yi et al. (2020). Our model clearly outperforms the previous state-of-the-art model. By analysis of our results, we observed that the last gap's prediction was PERIOD or QUESTION for all samples but one. Moreover, most of the errors are due to predicting a COMMA for a gap whose reference does not attribute any punctuation to, and vice-versa. This supports the notion of an underlying heuristic to solve the punctuation task for models trained at the sentence-level.

In our pipeline scenario, the main problem with this type of training is the implicit dependency on correct segmentation of the ASR outputs. If the text is not correctly segmented, the model will always predict a PERIOD or QUESTION tag for the last gap of a segment, either it makes sense linguistically or not. Moreover, training the models at the segment-level make them inapt to restore punctuation for samples which consist of multiple sentences. In Table 7, we report results that support this notion. We notice that the model trained at the sentence-level performs remarkably poorly when evaluated on longer samples consisting of multiple segments. Notice that the biggest impact on performance comes from the PERIOD and QUESTION labels, once the model is biased to predict these labels only at the end of each sample. In contrast, the model trained at the chunk-level is able to perform very satisfactorily either when it is evaluated at the segment-level or at the chunk-level. As Ueffing et al. (2013) reports, if we were to evaluate our model at the document-level, we would obtain a lower bound for the performance, and if we were to evaluate our model at the segment-level, we would obtain a upper bound for the performance. The truth lies in the middle – that is precisely our approach.

7.3.2. Error Propagation in the ASR Evaluation

We observed in Table 2 that the model's performance in the ASR track is considerably degraded when compared to the performance in the human-edited transcripts track. We briefly discussed these results in Section 7.1.1. In Table 6, we compare our best model trained at the segment-level with the best model reported in Yi et al. (2020) for the ASR track. We report the results for the evaluation at the segment-level as well. The results for our best model trained at the chunk-level and evaluated at the chunk-level can be found in Table 2. Notice that our model outperforms the model in Yi et al. (2020) for all labels except for the COMMA. This might be due to how we are transferring punctuation from the reference to the ASR output.

The work in Yi et al. (2020) cites back to Ueffing et al. (2013) in what comes to the corpora processing. As already mentioned in Section 4, in Ueffing et al. (2013), the authors opted to restrict the evaluation to those punctuation marks whose context words to the left and the right have been recognized correctly, as the transfer of punctuation based on the Minimum Edit Distance did not lead to satisfactory results because the utterances contained many ASR errors. However, we think this evaluation is not entirely suitable for our use-case, since the authors opted to alleviate a problem that is likely to actually occur in our pipeline. Note that this effect is not felt as much for the PERIOD label, once it is very likely that each sample contains a single PERIOD label located at the end of the sample. Thus, it will be transferred precisely, and subsequently predicted correctly by making use of the positional information of the model trained at the segment-level.

8. Conclusions

We have proposed an approach for punctuation prediction that achieves state-of-the-art results. Our architecture is composed of three main building blocks: a pre-trained Transformer-based encoder model that consists of either RoBERTa or XLM-RoBERTa, an attention mechanism over the encoder layers, and a multinomial logistic regression classifier. We have conducted our evaluation experiments both in the transcripts of TED Talks from the IWSLT 2017 evaluation campaign, and in the English transcripts of TED Talks from IWSLT 2011, allowing to further compare our results with related work reported in the literature.

We have applied our architecture first as a monolingual setting, and later as a multilingual one. Concerning the monolingual setting, we have created monolingual models for English, revealing that RoBERTa outperforms XLM-RoBERTa as the encoder model, a predictable result since RoBERTa was pre-trained only on English data and XLM-RoBERTa was pre-trained with multilingual data. We have also observed that the recognition errors of the ASR output degrade the performance of our models, in line with the related literature. The most frequent errors are due to situations where commas and periods can be used interchangeably. As our models do not rely in positional information to predict PERIOD and COMMA labels, the most frequent errors are predicting a COMMA for a given gap whose target reference is PERIOD, and vice-versa. We also have created individual monolingual models for Human-edited transcripts in German, Dutch, Portuguese and Romanian. All the models consistently performed better in the prediction of full-stop, revealing that COMMA may be more difficult to predict using pre-trained contextual models. Finally, we have trained a single multilingual model that can be used for predicting punctuation across multiple languages. The corresponding results are comparable with results achieved with monolingual models, even surpassing the existing results for 3 of the 5 languages. Such results evidence the potential of using a single multilingual model to solve the task for multiple languages. This is of particular importance for our video subtitling pipeline, since training a single model is more efficient than training and selecting a different model for each language. The achieved results also confirm that a morphologically rich language, such as the Portuguese, also fits into the concept of multilingual punctuation. Finally, it is important to refer that, to the best of our knowledge, there is not a study in the literature for punctuation restoration for multiple languages with a single model.

Finally, we have performed additional experiments to provide, as much as possible, a fair comparison with the existing state-of-the-art. We argue that usage of current punctuation systems in the literature implicitly require correct segmentation of ASR outputs for they rely on heuristics to solve the punctuation task. This is a major issue for incorporation of such models in a real life application. Through several experiments, we show that our method to train and test models is more robust to different segmentation.

In the future we plan to extend this work to include other language families, besides Germanic and Romance, such as Semitic and Slavic languages. That will allow us to strengthen our multilingual approach or, as punctuation for different languages may involve different feature sets (Szaszák and Ákos Tündik, 2019; Nanchen and Garner, 2019), help identifying possible limitations.

For reproducibility purposes, the developed code and respective hyperparameters are provided in the address: `https://github.com/Unbabel/caption`.

Acknowledgments

This work was supported by national funds through FCT, Fundação para a Ciência e a Tecnologia, under project UIDB/50021/2020 and by PT2020 funds, under the project "Unbabel Scribe: AI-Powered Video Transcription and Subtitle" with the contract number: 038510. This work was done while Nuno was interning at Unbabel. The authors have contributed equally to this work. The authors would like to thank the anonymous reviewers for their constructive comments.

References

K. Zechner, Automatic summarization of open-domain multiparty dialogues in diverse genres, Computational Linguistics 28 (2002) 447–485.

J. Huang, G. Zweig, Maximum entropy model for punctuation annotation from speech, in: Proc. of the ICSLP, 2002, pp. 917 – 920.

J.-H. Kim, P. C. Woodland, A combined punctuation generation and speech recognition system and its performance enhancement using prosody, Speech Communication 41 (2003) 563 – 577. doi:DOI: 10.1016/S0167-6393(03)00049-9.

M. Ostendorf, E. Shriberg, A. Stolcke, Human language technology: Opportunities and challenges, in: Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05), Philadelphia, 2005.

D. Jones, E. Gibson, W. Shen, N. Granoien, M. Herzog, D. Reynolds, C. Weinstein, Measuring human readability of machine generated text: three case studies in speech recognition and machine translation, in: Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05), volume 5, 2005, pp. v/1009–v/1012. doi:10.1109/ICASSP.2005.1416477.

J. Makhoul, A. Baron, I. Bulyko, L. Nguyen, L. Ramshaw, D. Stallard, R. Schwartz, B. Xiang, The effects of speech recognition and punctuation on information extraction, in: INTERSPEECH-05, 2005, pp. 57–60.

E. Shriberg, Spontaneous speech: How people really talk, and why engineers should care, in: Proc. of Eurospeech - 9^{th} European Conference on Speech Communication and Technology (Interspeech 2005), Lisbon, Portugal, 2005, pp. 1781 – 1784.

E. Matusov, A. Mauser, H. Ney, Automatic sentence segmentation and punctuation prediction for spoken language translation, in: International Workshop on Spoken Language Translation, Kyoto, Japan, 2006, pp. 158–165.

R. Cattoni, N. Bertoldi, M. Federico, Punctuating confusion networks for speech translation, in: Proc. of the 8^{th} Annual Conference of the International Speech Communication Association (Interspeech 2007), 2007, pp. 2453–2456.

M. Ostendorf, B. Favre, R. Grishman, D. Hakkani-Tür, M. Harper, D. Hillard, J. Hirschberg, H. Ji, J. G. Kahn, Y. Liu, S. Maskey, E. Matusov, H. Ney, A. Rosenberg, E. Shriberg, W. Wang, C. Wooters, Speech segmentation and spoken document processing, IEEE Signal Processing Magazine 25 (2008) 59–69.

J. Liao, S. E. Eskimez, L. Lu, Y. Shi, M. Gong, L. Shou, H. Qu, M. Zeng, Improving readability for automatic speech recognition transcription, 2020.

S. Peitz, M. Freitag, A. Mauser, H. Ney, Modeling punctuation prediction as machine translation, in: International Workshop on Spoken Language Translation, San Francisco, CA, USA, 2011, pp. 238–245.

S. Kowal, D. C. O'Connell, Communicating with One Another: Toward a Psychology of Spontaneous Spoken Discourse, Cognition and Language: A Series in Psycholinguistics, Springer New York, 2008.

S. M. Shieber, X. Tao, Comma restoration using constituency information, in: NAACL '03: Proc. of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Association for Computational Linguistics, Morristown, NJ, USA, 2003, pp. 142–148.

M. Harper, B. Dorr, J. Hale, B. Roark, I. Shafran, M. Lease, Y. Liu, M. Snover, L. Yung, A. Krasnyanskaya, R. Stewart, Parsing and spoken structural event detection, in: 2005 Johns Hopkins Summer Workshop Final Report, 2005.

J. Mrozinsk, E. W. Whittaker, P. Chatain, S. Furui, Automatic sentence segmentation of speech for automatic summarization, in: Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '06), 2006.

D. Beeferman, A. Berger, J. Lafferty, Cyberpunc: a lightweight punctuation annotation system for speech, ICASSP (1998) 689–692.

H. Christensen, Y. Gotoh, S. Renals, Punctuation annotation using statistical prosody models, in: Proc. of the ISCA Workshop on Prosody in Speech Recognition and Understanding, 2001, pp. 35–40.

J. Kim, P. C. Woodland, The use of prosody in a combined system for punctuation generation and speech recognition, in: Proc. of Eurospeech, 2001, pp. 2757–2760.

Y. Liu, E. Shriberg, A. Stolcke, D. Hillard, M. Ostendorf, M. Harper, Enriching speech recognition with automatic detection of sentence boundaries and disfluencies, IEEE Transaction on Audio, Speech and Language Processing 14 (2006) 1526–1540.

F. Batista, D. Caseiro, N. J. Mamede, I. Trancoso, Recovering punctuation marks for automatic speech recognition, in: Interspeech 2007, Antwerp, Belgium, 2007, pp. 2153 – 2156.

F. Batista, D. Caseiro, N. Mamede, I. Trancoso, Recovering capitalization and punctuation marks for automatic speech recognition: Case study for portuguese broadcast news, Speech Communication 50 (2008) 847–862.

F. Batista, I. Trancoso, N. J. Mamede, Comparing automatic rich transcription for portuguese, spanish and english broadcast news, in: Automatic Speech Recognition and Understanding, 2009. ASRU 2009. IEEE Workshop on, IEEE, 2009, pp. 540–545.

W. Lu, H. T. Ng, Better punctuation prediction with dynamic conditional random fields, in: EMNLP, 2010.

F. Batista, H. Moniz, I. Trancoso, H. Meinedo, A. I. Mata, N. J. Mamede, Extending the punctuation module for european portuguese., in: Interspeech 2010, Mukari, Japan, 2010, pp. 1509–1512.

F. Batista, H. Moniz, I. Trancoso, N. J. Mamede, Bilingual experiments on automatic recovery of capitalization and punctuation of automatic speech transcripts, IEEE Transactions on Audio, Speech and Language Processing, Special Issue on New Frontiers in Rich Transcription 20 (2012) 474–485. doi:10.1109/TASL.2011.2159594.

N. Ueffing, M. Bisani, P. Vozila, Improved models for automatic punctuation prediction for spoken and written text, in: INTERSPEECH, 2013.

Y. Liu, E. Shriberg, A. Stolcke, B. Peskin, J. Ang, D. Hillard, M. Ostendorf, M. Tomalin, P. Woodland, M. Harper, Sructural metadata research in the EARS program, in: Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05), Philadelphia, USA, 2005.

S. Stüker, C. Fügen, R. Hsiao, S. Ikbal, F. Kraft, M. Paulik, M. Raab, Y.-C. Tam, M. Wölfel, The ISL TC-STAR spring 2006 ASR evaluation systems, in: Proc. of the TC-STAR Workshop on Speech-to-Speech Translation, Barcelona, Spain, 2006.

O. Tilk, T. Alumäe, Lstm for punctuation restoration in speech transcripts, in: INTERSPEECH, 2015.

O. Tilk, T. Alumäe, Bidirectional recurrent neural network with attention mechanism for punctuation restoration, in: INTERSPEECH, 2016, pp. 3047–3051.

X. Che, C. Wang, H. Yang, C. Meinel, Punctuation prediction for unsegmented transcript based on word vector, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), European Language Resources Association (ELRA), Portorož, Slovenia, 2016, pp. 654–658. URL: https://www.aclweb.org/anthology/L16-1103.

O. Klejch, P. Bell, S. Renals, Sequence-to-sequence models for punctuated transcription combining lexical and acoustic features, 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017) 5700–5704.

J. Yi, J. Tao, Self-attention based model for punctuation prediction using word and speech embeddings, ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019) 7270–7274.

S. Kim, Deep recurrent neural networks with layer-wise multi-head attentions for punctuation restoration, ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019) 7280–7284.

P. Zelasko, P. Szymanski, J. T. Mizgajski, A. Szymczak, Y. Carmiel, N. Dehak, Punctuation prediction model for conversational speech, in: INTERSPEECH, 2018.

C. Cieri, D. Miller, K. Walker, The fisher corpus: a resource for the next generations of speech-to-text, in: Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04), European Language Resources Association (ELRA), Lisbon, Portugal, 2004. URL: http://www.lrec-conf.org/proceedings/lrec2004/pdf/767.pdf.

A. Nanchen, P. N. Garner, Empirical evaluation and combination of punctuation prediction models applied to broadcast news, ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019) 7275–7279.

M. Federico, L. Bentivogli, M. Paul, S. Stüker, Overview of the IWSLT 2011 evaluation campaign, in: 2011 International Workshop on Spoken Language Translation, IWSLT 2011, San Francisco, CA, USA, December 8-9, 2011, ISCA, 2011, pp. 11–27.

J. Yi, J. Tao, Z. Wen, Y. F. Li, Distilling knowledge from an ensemble of models for punctuation prediction, in: INTERSPEECH, 2017.

J. Yi, J. Tao, Y. Bai, Z. Tian, C. Fan, Adversarial transfer learning for punctuation restoration, 2020. arXiv:2004.00248.

J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. arXiv:1810.04805.

T. Pires, E. Schlinger, D. Garrette, How multilingual is multilingual BERT?, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 4996–5001. URL: https://www.aclweb.org/anthology/P19-1493. doi:10.18653/v1/P19-1493.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. arXiv:1907.11692.

Y. Cai, D. Wang, Question mark prediction by bert, in: 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2019, pp. 363–367.

K. Makhija, T. Ho, E. Chng, Transfer learning for punctuation prediction, in: 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2019, pp. 268–273.

O. Klejch, P. Bell, S. Renals, Punctuated transcription of multi-genre broadcasts using acoustic and lexical approaches, in: 2016 IEEE Spoken Language Technology Workshop (SLT), 2016, pp. 433–440. doi:10.1109/SLT.2016.7846300.

P. Bell, M. J. F. Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester, P. C. Woodland, The MGB challenge: Evaluating multi-genre broadcast media recognition, in: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2015, Scottsdale, AZ, USA, December 13-17, 2015, IEEE, 2015, pp. 687–693. URL: https://doi.org/10.1109/ASRU.2015.7404863. doi:10.1109/ASRU.2015.7404863.

M. Ákos Tündik, B. Tarján, G. Szaszák, A low latency sequential model and its user-focused evaluation for automatic punctuation of asr closed captions, Computer Speech & Language 63 (2020) 101076. URL: https://www.sciencedirect.com/science/article/pii/S0885230820300097. doi:https://doi.org/10.1016/j.csl.2020.101076.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, 2017. arXiv:1706.03762.

R. Sennrich, B. Haddow, A. Birch, Neural machine translation of rare words with subword units, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Berlin, Germany, 2016, pp. 1715–1725. URL: https://www.aclweb.org/anthology/P16-1162. doi:10.18653/v1/P16-1162.

I. Tenney, D. Das, E. Pavlick, Bert rediscovers the classical nlp pipeline, 2019. arXiv:1905.05950.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, 2018. arXiv:1802.05365.

D. Kondratyuk, M. Straka, 75 languages, 1 model: Parsing universal dependencies universally, 2019. URL: https://www.aclweb.org/anthology/D19-1279. arXiv:1904.02099.

J. Howard, S. Ruder, Universal language model fine-tuning for text classification, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 328–339. URL: https://www.aclweb.org/anthology/P18-1031. doi:10.18653/v1/P18-1031.

G. Szaszák, M. Ákos Tündik, Leveraging a Character, Word and Prosody Triplet for an ASR Error Robust and Agglutination Friendly Punctuation Approach, in: Proc. Interspeech 2019, 2019, pp. 2988–2992. URL: http://dx.doi.org/10.21437/Interspeech.2019-2132. doi:10.21437/Interspeech.2019-2132.

A. Nanchen, P. N. Garner, Empirical evaluation and combination of punctuation prediction models applied to broadcast news, in: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 7275–7279. doi:10.1109/ICASSP.2019.8683796.