Leveraging Prosody for Punctuation Prediction of Spontaneous Speech

Jenny Yeonjin Cho
조연진

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2022

Committee:
Mari Ostendorf
Gina Levow

Program Authorized to Offer Degree:
Electrical Engineering

University of Washington

Abstract

Leveraging Prosody for Punctuation Prediction
of Spontaneous Speech

Jenny Yeonjin Cho

조연진

Chair of the Supervisory Committee:
Professor Mari Ostendorf
Electrical and Computer Engineering

Clarity and precision of written text benefits from correct punctuation. For scripts that lack punctuation, such as conversational speech, there can be errors in accurately interpreting the intention of a speaker based on the words only. There have been efforts in the past to predict punctuation using a variety of language models, but such studies have not taken full advantage of prosody in a neural language model. Several studies have found simple pauses to be a useful method to capture some punctuation marks, but not all punctuation marks are associated with a pause. There are no recent studies that make use of all available prosodic correlates; thus, I explore the benefit of using intonation and energy in addition to the simple pauses. This thesis aims to bridge the gap between recent work and prosody by introducing a new neural model for punctuation prediction that incorporates various prosodic features, such as pauses, duration, pitch and energy of speech. The goal is to improve automatic punctuation prediction in transcriptions of spontaneous speech. In addition, I pose the question of how to represent interruption points—when a speaker breaks the standard grammatical flow of a sentence to repeat or correct a phrase—associated with disfluencies in spontaneous speech. In various experiments on the Switchboard corpus, I find that prosodic information improves punctuation prediction fidelity for both hand transcripts and automatic

speech recognition output. The word errors present in the automatic transcriptions hinder the punctuation prediction results at a rate that roughly corresponds to its word error rate. I find that automatically transcribed scripts with word errors benefit more from taking advantage of all prosody features than hand transcripts do. I also find that explicit modeling of interruption points benefits the performance for standard punctuation sets, and that it is better to represent them as commas than no punctuation.

TABLE OF CONTENTS

LIST OF FIGURES

GLOSSARY

ASR: automatic speech recognition

BERT: Bidirectional Encoder Representations from Transformers

CNN: convolutional neural network

DA: dialogue act

GRU: gated recurrent units (a type of RNN)

HMM: hidden Markov model

LSTM: long short term memory (a type of RNN)

MFCC: mel-frequency cepstral coefficients

NLP: natural language processing

RNN: recurrent neural network

SWBD: Switchboard (corpus)

WER: word error rate

ACKNOWLEDGMENTS

This final piece of work for my master's degree would not have been possible without the support I received from the people around me. First, I would like to acknowledge my collaborators, Sara Ng and Trang Tran, who made this thesis possible. This work is a joint effort with Sara. After I ran different experiments to finalize the hyperparameters and the features extraction process, she helped with inference tasks using the final models and debugging errors that came up during evaluation.

I would like to thank Professor Mari Ostendorf who accepted me as her Master's student despite her busy schedule. Her standard of excellence and trust in my ability to do well has pushed me to reach for more things that I never saw myself doing (like this thesis...). I would also like to thank Ellen, Kevin, Michael, Roy, and Sitong (and Yushi!) for helping me adjust and find my way around TIAL lab.

Finally, I am thankful for my parents, my biggest supporters from the beginning. I am where I am now because of their silence sacrifices.

DEDICATION

to my Heavenly Father,

...

and to my high school self,

who wasn't planning to go to college

...but now finishing a MS degree with a thesis.

...

(well, well... how the turntables)

Chapter 1

INTRODUCTION

Punctuation is a crucial part of many languages, such as English, as inappropriate placement or use of punctuation marks can semantically alter the intention of a writer. Punctuation marks are not verbalized in spoken language, and sentence structure is communicated via prosodic cues, such as pauses, duration lengthening, and intonation associated with the phrase structure and utterance intent. In written text, punctuation serves as a proxy for prosodic information. On the flip side, when spontaneous speech is transcribed by an automatic speech recognition (ASR) systems, the punctuation marks that one would expect in written text are not captured, resulting in transcripts that can be difficult to comprehend correctly. This is especially true when lack of punctuation is compounded with automatic transcription errors. Automatic prediction of *punctuation* for transcribed speech is therefore important to correctly represent the structure of the spoken utterance.

In formal contexts or read speech, punctuation can be predicted reasonably well from word context alone (i.e. without attending to prosodic features), particularly when modeled by powerful neural language models. Perhaps for that reason, most recent punctuation prediction work does not take advantage of prosody. However, for conversational speech, which does not adhere to written grammatical structure and often includes disfluencies, it may be that prosodic cues are more helpful in intent comprehension. In addition, prosodic cues may help compensate for ASR errors, though the prosodic features themselves may be sensitive to ASR errors. In many cases, pauses are reasonably reliable when predicting punctuation, but speakers can use pauses for multiple reasons, including hesitation and in putting special emphasis on a word. Moreover, punctuation is not always associated with a pause.

My work in this thesis explores a mechanism for incorporating prosodic features into a

neural punctuation prediction model, building on prior work in spoken language processing. Looking at conversational speech specifically, I explore three questions. First, to what extent does prosody (beyond just pauses) improve punctuation prediction? Addressing this question involves exploring how various prosodic features that are more than simple pauses might impact a model's performance. Second, we explore explicit labeling and prediction of interruption points associated with disfluencies. In brief, interruption points refer to when a speaker introduces a break in their own speech to repeat or correct a phrase. There is no standard convention for punctuation associated with interruption points and it is often left unmarked. What confusions might this resolve or introduce, and could this be helpful in predicting other definite punctuation marks if interruption points are explicitly represented in the punctuation set? Lastly, to what extent do ASR word errors impact the prediction performance? This question explores how incorrect words may hinder the prediction performance and whether prosody can play a role in improving the performance.

The rest of the thesis is outlined as follows. Chapter 2 presents the motivation for punctuation prediction, reviews fundamental concepts that this work builds on, and discusses how this thesis goes beyond previous related work. Chapter 3 describes the main task along with the models and ASR system used to tackle this task. Chapter 4 outlines the experimental set up and discusses the results of the experiments together with analysis. Chapter 5 concludes the thesis with a summary and potential directions for future work.

Chapter 2
BACKGROUND

This chapter describes the prior studies and the related work that this thesis builds on in order to provide context for interpreting the results included in Chapter 4. The first half defines prosody and motivates its relevance to punctuation and neural sequence models. Then the section is followed with a brief overview of the most common neural sequence models that are used. The last section explores the related work in punctuation modeling as well as the use of prosody to enhance performance of this task.

2.1 Prosody and Disfluencies

Prosody is a way to express speaker intent in addition to the identity and order of the spoken words. It adds semantic significance, helps resolve syntactic ambiguities, and allows speech to be more expressive. Prosody can be defined in two ways: (1) by *function* and (2) by *form* (Wagner and Watson, 2010). When defined by its function, the word refers to properties of speech that are not necessarily related to the spoken words' definition, but more closely related to elements like rhythmic grouping or emphasis placement on the words, and attitude of speakers. When defined by its form, it refers to acoustic attributes that can be measured. The main attributes of prosody include varying pitch (fundamental frequency), duration, and intensity (energy). In this work, *prosody* refers to both definitions; the function definition refers to the segmentation of marking punctuation, and the form definition refers to acoustic feature extraction from speech signals.

Duration captures word lengthening within words, and pauses capture the time between words. Fundamental frequency captures pitch excursions at word boundaries and relative accents between words. The boundary tones, which are pitch changes that happen at the edge

of prosodic events, are known to align to the end or beginning of words. This fact is important for an experimental decision made in Chapters 3 and 4. Without varied pitch, duration, or emphasis in spoken language, speaker intention can potentially suffer from ambiguity and monotony. As there are no means of explicitly marking prosody on text, punctuation is a way to communicate some of the meaning associated with prosody to readers. The prosody features used in this work can be found in Section 3.2.1.

In linguistics, *disfluencies* refer to when flow of speech is interrupted. Such interruptions can happen for various reasons such as repeating words, self correcting previously spoken words, restarting a sentence, hesitating with a silent pause or filling the pause with *uh* and *um* (Clark and Fox Tree, 2002; Lickley, 2015). Disfluencies are not the same as speaker errors which is when the speaker says an incorrect word but does not interrupt the flow of speech. Although disfluencies interrupt the flow of speech, comprehension is not necessarily impaired. Disfluencies are a natural part of conversational speech and they happen constantly, which is why it is important to look at how these should be perceived by a punctuation prediction model. In this work, I explore the question of how punctuation prediction benefits or does not benefit from explicitly marking disfluency interruption points associated with repetitions, self-corrections, and restarts.

2.2 Neural Sequence Models

Sequence models are computational or mathematical models that input and/or output a stream of data, or a sequence. Some examples of input data are audio signals (e.g., for speech recognition), video clips (e.g., activity recognition), and sequences of text (e.g., sentiment analysis). The sequential nature of language makes sequence models favorable for natural language processing (NLP) tasks. Sequential models can be unidirectional or bidirectional. A key difference is that unidirectional models can only access information from the past, whereas bidirectional models can access both past and future information; one version of the input seen from past to future and another seen from future to past. This is important for many language tasks since word ambiguities exist. In this work, I use three main classes of

neural networks: recurrent neural networks (RNN), transformers, and convolutional neural networks (CNN).

2.2.1 Recurrent Neural Networks

A common class of neural sequence model, RNN, is an umbrella term that refers to different models with the recurrent nature. These are a bit different from the traditional feed-forward network Within RNNs there is the long-short term memory (LSTM) (Hochreiter and Schmidhuber, 1997), and gated recurrent units (GRU) (Cho et al., 2014). Vanilla RNNs suffer from vanishing gradient issues. Gradient values, which are used in updating the neural network's weights during back propagation, can diminish as the updating process propagates back in time, which can lead to insignificant contribution to the learning steps. This means that in long sequences, the model may have trouble making use of earlier time step information. LSTMs and GRUs were invented to work around the vanishing gradient issue. They both use gating mechanisms to control the information that flows through the neural cell sequence so that the short-term memory issue is mitigated. The main differences between a LSTM and GRU are that GRUs are simpler in structure (GRU has 2 gates vs. LSTM has 3 gates), and GRUs can control the flow of information without using a memory unit or a cell state. In this thesis, I use RNNs, specifically bidirectional GRUs, to predict the punctuation labels. This decision follows the dialogue act (DA) recognition work in (Tran, 2020).

2.2.2 Transformers

Another popular model is a transformer which was initially developed for a machine translation task (Vaswani et al., 2017). An advantage of transformers is that these can process the input as whole sentences instead of word by word, allowing for parallelization in training and thus speeding up the task. Another advantage is that transformers have a concept of self-attention, which is a method used to contextualize the word representation. In the hidden layers of a transformer, the layers can create a representation of the input by relating different words of the same sentence. This mechanism allows the inputs to have a richer

context in relation to their surrounding words. This work uses a transformer-based model called Bidirectional Encoder Representations from Transformers (BERT), which is a powerful state-of-the-art language model that can be fine-tuned for domain adaptation (Devlin et al., 2019). To pretrain BERT, two tasks are used: (1) Masked Language Model and (2) next sentence prediction. Masked language model prediction is where a portion of a sentence is covered and the model learns to predict those words correctly. This process enables training of embeddings in a bidirectional manner as the guessed words are based on the words to the left and right of them. Next sentence prediction is when the model learns to distinguish whether the next appearing sentence follows the first, or is completely random. Pretrained BERT can be easily fine-tuned to adapt to the task in interest. In this thesis, the transformer is used to contextualize the word vectors input to the RNN.

2.2.3 Convolutional Neural Networks for Feature Extraction

A convolutional neural network (CNN) is a machine learning model that has been explored by many scholars for decades (Zeiler and Fergus, 2013). A CNN builds on two simple operations, convolution and pooling, combining multiple filters at different scales. Currently, CNNs dominate the image-related machine learning fields. Recent exciting work with CNNs include face recognition, object detection, classification, and segmentation, video processing, natural language (text and speech) processing, and many more. In this work, a CNN is used for processing frame-based prosody features given an audio signal of the conversation, which follows what was done in (He et al., 2018).

2.3 Related Work in Punctuation Modeling

To increase readability for people as well as automatic language processing systems, researchers have been exploring methods for automatic punctuation prediction for many years. Some of the early works use a trigram/n-gram probability model (Beeferman et al., 1998; Gravano et al., 2009), statistical model (Christensen et al., 2001), and a max entropy model (Huang and Zweig, 2002). Recent work has transitioned from statistical models to leverage

neural models. Several different variants of RNNs are explored in (Zelasko et al., 2018; Klejch et al., 2017), and CNNs are also used in (Zelasko et al., 2018). Recent studies primarily use pre-trained transformers (Makhija et al., 2019; Pappagari et al., 2021), which is an approach taken in this thesis along with GRUs. A comparison of the different model architectures can be found in (Sunkara et al., 2020).

Most studies, including the aforementioned ones, have been based on English. There have been some attempts to recover punctuation in other languages (see Moró and Szaszák (2017) for Hungarian and Fang et al. (2019); Guo et al. (2010); Zhao et al. (2012) for Chinese). The work on English has involved a variety of speech styles, including audio books (Levy et al., 2012; Pappagari et al., 2021), broadcast speech (Klejch et al., 2017), TED Talks (Makhija et al., 2019), medical dictation (Sunkara et al., 2020), and conversational speech (Zelasko et al., 2018; Sunkara et al., 2020; Pappagari et al., 2021). Out of these styles, this thesis focuses on conversational speech.

Currently, there is no standard set of punctuation marks used in punctuation prediction tasks. The most common set consists of {comma, period, question mark}, which is used in two of the conversational speech studies. However, Pappagari et al. (2021) predict punctuation marks from a larger set that includes {full stop, comma, question mark, exclamation mark, semicolon, double-dash, ellipsis}. In this thesis, I consider the expansions of the common punctuation set to handle two conversational speech phenomena: incomplete sentences and disfluent interruption points. Interruption points are often not explicitly labeled with a punctuation mark. Differences in training data and output punctuation tagsets make it difficult to compare results from different neural punctuation models. The quality and size of training data, as well as its similarity to written language, will impact a neural punctuation model's ability to generalize. As conversational speech data comes with more complicated attributes, such as disfluencies, many studies only consider hand transcripts; results on automatic transcripts are presented in Klejch et al. (2017); Sunkara et al. (2020), both showing lower F1 scores for ASR.

While several early studies explore the use of prosodic features in punctuation prediction

(Christensen et al., 2001; Huang and Zweig, 2002; Levy et al., 2012), most recent work relies solely on the speech transcripts. A notable exception is (Klejch et al. (2017)), which leverages features similar to the work in this thesis, but within a hierarchical RNN framework. They explore using only text, only prosody features, and both text and prosody, and they found that using all features (text, filterbank, and pitch) yields the best result. The key difference in this approach is the neural architecture (transformer + RNN + CNN) and inclusion of pause and duration features.

2.4 Related Work on Using Prosody in Spoken Language Processing

Prosody has been used in many spoken language processing tasks, most notably in segmentation, parsing, disfluency detection, and DA recognition. These studies inform the punctuation prediction work in terms of how prosody can be integrated with text, because they all leverage the constituent boundary marking function of prosody. In a long line of work, prosody was shown to improve topic segmentation (Hirschberg and Nakatani, 1998; Tür et al., 2001), sentence boundary detection (Kim and Woodland, 2003; Liu et al., 2004; Kolář et al., 2006), and turn segmentation (Hirschberg et al., 2004). Kahn et al. (2005) leveraged automatically predicted prosodic labels (i.e. ToBI (Silverman et al., 1992)) in a statistical parser, showing improvements in both parsing and disfluency detection. Similarly, in (Kahn and Ostendorf, 2012), prosody was shown to benefit joint parsing, sentence segmentation, and word recognition, especially when sentence boundaries were unknown. In more recent work on parsing with neural models, Tran et al. (2018) modeled raw acoustic features and showed the benefit of prosody especially in disfluent sentences and attachment error corrections, assuming known sentence boundaries.

Of these different tasks, dialog act recognition is most relevant to punctuation prediction. Most work in DA recognition focused on classification of a DA given a known (hand-segmented) utterance. In work with Hidden Markov models (HMM), the use of prosody was shown to be beneficial, specifically in distinguishing questions from statements, and backchannels from agreements (Shriberg et al., 1998; Stolcke et al., 2000). Prosody is shown

to be useful in domain adaptation for recognizing a simplified dialogue act set (statement, question, backchannel, incomplete), with gains mainly for questions (Margolis et al., 2010). Using a neural approach similar to (Tran et al., 2018), Tran (2020) showed that prosody benefited joint segmentation and DA classification, where prosody and dialog history seem to be complementary—prosody benefits segmentation while history benefits classification. The work in this thesis will be based on (Tran and Ostendorf, 2021), which is described further in the next chapter.

Many of these studies, however, relied on hand transcripts. For ASR outputs, Stolcke et al. (2000) showed that HMM-based DA and ASR systems benefit from joint recognition. Additionally, He et al. (2018) applied a CNN on segment-level MFCCs, and improved accuracy for experiments using only ASR outputs. A joint DA segmentation and classification system with an acoustic-to-word model is described in (Dang et al., 2020), but it was unclear where performance most suffered by using imperfect transcripts.

This thesis builds on the CNN approach to integrating prosodic features developed by Tran (2020). This model has shown to work well for both parsing and on DA recognition tasks, and the benefit of prosody extended to ASR transcripts. In this thesis, we look at whether it extends to punctuation prediction.

Chapter 3

METHODS

This chapter describes the main task of the thesis work. The dataset, prediction model architecture, and ASR system used to find the automatic transcriptions for the punctuation prediction task are described in this chapter.

3.1 Task and Data

The main task of this work is to predict punctuation labels for each word in the input. This task is similar to other natural language tasks such as named entity recognition which uses inside-outside-beginning tags for each word. The dataset used in this work is Switchboard (SWBD) (Godfrey and Holliman, 1993), which is a collection of spontaneous telephone speech between strangers who are prompted to talk about a specific set of topics. SWBD has been widely used for a number of speech understanding tasks including parsing, disfluency detection, dialog act recognition, sentence segmentation, and speech recognition. This thesis builds on a system originally designed for joint dialog act segmentation (Zhao and Kawahara, 2019) and recognition, so the portion of SWBD that was annotated with dialog acts is used for the punctuation task. For training, tuning, and testing of the different models, the data split commonly seen in dialog act classification is used, which is defined in (Jurafsky et al., 1997).

The standard punctuation set used in most work consists of {period (P.), question mark (Q?), comma (C,)}. In this work, we augment this set by a marker for an incomplete sentence (Inc-). As part of the questions to be explored, we incorporate the interruption point (IP+) as an additional category. Tables 3.1 and 3.2 show statistics of the dataset.

Table 3.1: Data statistics of SWBD. The disfluencies column indicates the percentage of dialogues that contain disfluency annotations.

Split	# Dialogues	# Turns	# Sentences	# Tokens	Disfluencies
train	1.1K	107K	194K	1.4M	100%
dev	21	1.6K	3.2K	25K	0%
full test	19	2.4K	4.1K	29K	74%
IP test	14	1.7K	2.9K	21K	100%

Table 3.2: Count of punctuation in each split. 'C,'=comma; 'P.'=period; 'Inc-'=incomplete; 'Q?'=question; 'O'=no punctuation; 'IP+'=interruption

Split	C,	P.	Inc-	Q?	O	IP+	Total
train	128K	127K	9.0K	7.8K	1.1M	48K	1.4M
dev	3.2K	2.2K	144	92	19K	68	25K
full test	2.8K	2.7K	175	197	22K	810	29K
IP test	1.8K	1.9K	134	125	16K	810	21K

3.2 Disfluency Annotations

There are multiple transcriptions of the SWBD data. This thesis uses the transcripts associated with disfluency annotations when available in order to investigate the effects of including interruption points as a punctuation category. However, the utterance times associated with the more careful Mississippi State transcriptions are used, which have been aligned to the earlier transcripts used in dialog act and disfluency annotations. The evaluation using interruption points is constrained to the available disfluency annotations. When disfluency annotations are unavailable for the training data, the default action was to only use the Switchboard annotations in (Jurafsky et al., 1997). Because of that there are two types of test sets, `full test` and `IP test`, which allows us to isolate the data with disfluency annotations. The percentage distribution of disfluency annonations are written in Table 3.1.

3.2.1 Data Preprocessing

The snippet below is an example from Figure 3.1, which we use to detail the data processing used to derive reference punctuation annotation. The following describes the text annotations. Note that this annotation is specific to the SWBD disfluency annotations.

```
You get a [ lot of, + ] {F uh,} {D you know, } great variety of things here,
/ {C so. } -/ {C But } if you were going to a restaurant, { D say, } {F um, }
where would you go? /
```

- { `F ...` } are filled pauses. These are floor holders in mid-conversation. (Ex. "uh", "um")

- { `C ...` } are conjunctions. (Ex. "but", "and", "so")

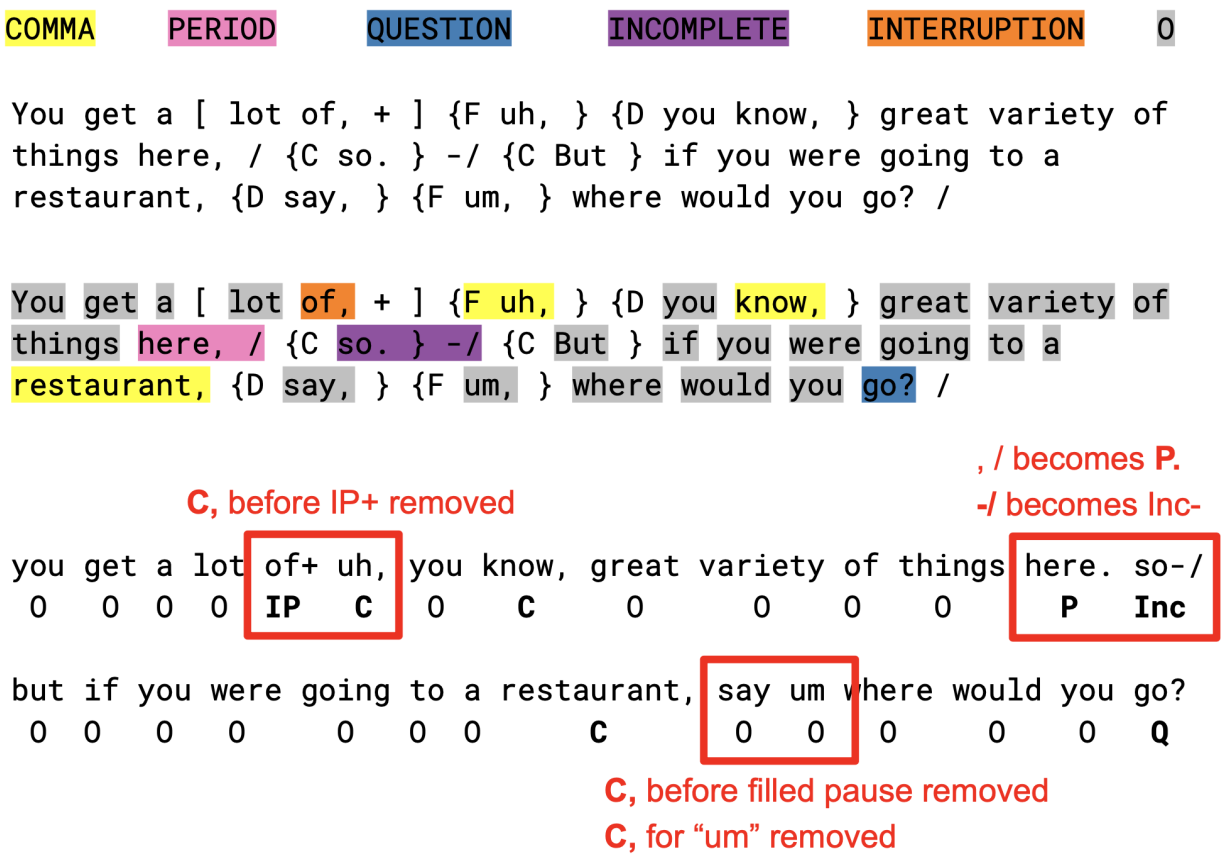


Figure 3.1: Example of text preprocessing. The top figure shows the raw data; The middle figure shows how punctuation tags are labeled; the bottom figure shows the result with annotations.

- { D ... } are discourse markers. These are catch-all markers that are often associated with starts of sentences. (Ex. "well", "you know")

- { E ... } are edit terms. These are used to correct oneself, sometimes after an interruption point.

- < ... > marks nonverbal utterances. (Ex. laughter, coughs)

- ((...)) are used to mark spoken words with uncertainty of what the word is supposed to be.

- # ... # are used to mark overlapped speech between two speakers.

- [... + ...] are used to mark disfluencies, that are often used to (1) repeat, (2) correct, or (3) give up and/or start something new. The + marks the interruption point.

- / marks the end of a sentence.

For the punctuation task, a sample is a speaker turn, which is the concatenation of successive utterances from a speaker up until the other speaker takes the floor (ignoring overlap for backchannels). Backchannels are treated as a full turn. Utterances that have no words (e.g. laughter) are removed. For speech recognition, turn times are based on the start and end times of the first and last utterance, respectively. Unlike many text-based models of SWBD, contractions are not separated into two tokens. This convention is more consistent with speech transcription conventions and the fact that punctuation never in occurs inside a contraction.

The ground truth punctuation labels are extracted from SWBD with some modifications. The original transcription convention includes commas around all filled pauses and at interruption points. All such commas were removed as this labeling is non-standard and artificially biases the predictions. The slash (/) marks the boundaries of sentence-like units,

so it is associated with a period irrespective of the punctuation used, assuming missing punctuation is an annotation error. Specific implementation of the aforementioned modifications and other mappings are listed below. Figure 3.1 illustrates an example of the process.

- Nonverbal transcriptions are removed.
- Commas preceding filled pauses (`uh` and `um`) are removed.
- Commas following filled pauses are removed, except before `you know`.
- All annotation tags within {}s are moved.
- Exclamation points (`!`), ellipses (`...`), and words followed by a plain slash are converted to periods.
- Periods are assigned to the word preceding `,/`, `./`, and `/`.
- Interruptions are assigned to the word preceding `+`.
- Incompletes are assigned to the word preceding `-/`.

3.2.2 Evaluation Metrics

Punctuation is evaluated using macro F-scores. For the case when the transcripts are automatically generated via an ASR, there may be words inserted or deleted. As there is no standard process for aligning reference punctuation to automatically transcribed text, I follow the general methodology described in Sunkara et al. (2020); if the automatic transcription inserts a word and that word is assigned a punctuation *and* the punctuation of the previous word in the aligned reference transcription matches the inserted word's punctuation, then it is considered correct.

First, the ASR and reference transcripts are aligned to minimize the Levenshtein distance. When the ASR predicts extra tokens, a dummy `<INSERTED>` token is added to the reference text, with tag `O` (no punctuation). Where the ASR misses or deletes a word, the dummy tag

<MISSED> is added to the ASR transcript, along with tag O for the punctuation mark. The prediction at index i is considered a true positive if and only if:

1. reference[i] $\neq$ <INSERTED>, ASR[i] $\neq$ <MISSED>, and reference_punct[i] = ASR_punct[i], or

2. $i \neq 0$, ASR[i] = <MISSED>, reference_punct[$i-1$] not in {.,}, ASR[$i-1$] $\neq$ <MISSED>, and reference_punct[i] = ASR_punct[i]

The equations in (3.1) show how the precision and recall are computed for the prediction of question mark ("?") in automatically generated transcripts:

$$P = \frac{|TP(?)|}{|? \text{ in ASR}|} \quad , \quad R = \frac{|TP(?)|}{|? \text{ in reference}|} \tag{3.1}$$

3.3 Prosody Features

The features described below are summarized from (Tran, 2020).

Pause. There are two ways that pauses are captured. From the raw pause duration p for an entire sequence, I create a pause feature vector which concatenates pre- and post-word pauses between words. Each value in the feature vector is one of 6 categorical pause features. The categories are defined below:

$$\text{pause categories} = \begin{cases} 1 & \text{no pause} \\ 2 & \text{missing}^1 \\ 3 & 0 < p \leq 0.05s \\ 4 & 0.05 < p \leq 0.2s \\ 5 & 0.2 < p \leq 1s \\ 6 & p > 1s \end{cases} \tag{3.2}$$

[1]This occurs when there are missing word boundary times. Such time alignments were used to compute the pause duration, 1% of sentences were missing times.

The bins have been selected based on the distribution of pause duration in the data.

The raw pause features are captured in a 2-dimensional vector with the normalized pre- and post-word pause lengths:

$$\text{pause}_{(pre|post),j} = min(1, ln(1 + p_{(pre|post),j}) \tag{3.3}$$

Word duration. This feature is a concatenated 2-dimensional vector with globally and locally normalized word duration. Global normalization is computed as $min(5, \frac{w_j}{\mu_j})$, where w_j is the duration of the word at index j, and μ_j is the average duration of the word type in the corresponding data set. There is a minimum threshold of 5 to control misalignment errors that may cause abnormally long durations. Local normalization is computed as $\frac{w_j}{max(W_i)}$, where W_i is a list of durations for turn i, and $max(W_i)$ is the maximum word duration of all words in utterance or turn i.

Filterbank. The energy feature is a 3-dimensional vector with the following: the log of total energy normalized by the speaker's max total energy, the log of total energy in the lower 20 bands normalized by the total energy, and log of total energy in the higher 20 bands normalized by the total energy. All contour features are extracted from the Kaldi 40-mel-frequency filterbank features (Povey et al., 2011). Each measure is computed over 25 ms frames with a 10 ms hop.

Pitch. The fundamental frequency contour features are also extracted from 25 ms frames with a 10 ms hop using Kaldi (Povey et al., 2011). This is a 3-dimensional feature vector consisting of warped Normalized Cross Correlation Function (NCCF), log-pitch normalized by probability of voicing, and the delta log of the pitch. (See bottom 3 acoustic features signal in Figure 3.2.) Each measure is computed over 25 ms frames with a 10 ms hop.

3.4 Model

The punctuation model is based on the dialog act recognition model from Tran (2020). Specifically, this model is an extension of the best performing RNN encoder-decoder model with attention in (Zhao and Kawahara, 2019), combined with the CNN module for learning acoustic-prosodic features as described in (Tran et al., 2018). Briefly, the encoder-decoder model takes a turn u as input, and each turn is represented by $x = [x_1, \ldots, x_T]$. The model learns to output the punctuation sequence $y = [y_1, \ldots, y_T]$ of a turn, where x_i is the word-level feature vector input, y_i is the punctuation label output, and T is the total sequence length.

Unlike in (Zhao and Kawahara, 2019), we do not use previous turn context labels, since punctuation prediction relies less on previous turns than dialogue act prediction. Given an RNN encoder that produces the hidden states $h_1, \ldots, h_T$, the RNN decoder computes $d_t = RNN([\tilde{y}_{t-1}; c_{t-1}], d_{t-1})$ where $\tilde{y}_{t-1}$ is the embedding associated with label y_{t-1} and $c_{t-1} = \sum_{i=1}^{T} \alpha_{t_i} h_i$, $\alpha_t = \text{softmax}(u_t)$, $u_t = v^t \tanh(W_1 h_i + W_2 d_t + b_a)$. The predicted signal y_t is determined by $p(y_t|h, y < t) = \text{softmax}(W_s[c_t; d_t] + b_s)$ A complete overview of the model is presented in Figure 3.2.

For the model which uses all prosody features, the input vectors $x_i = [e_i; \phi_i; s_i]$ are composed of word embeddings e_i, pause- and duration-based features ϕ_i, and learned energy/pitch (E/f0) features s_i, which taken together represent a prosodically contextualized word vector. The word embeddings e_i are pretrained BERT embeddings (Devlin et al., 2019) (BERT-base-uncased version), which have been shown to perform well on a variety of NLP tasks. Pause- and duration-based features ϕ_i are composed of both raw and categorical pause durations after each word; word durations are normalized by the mean duration of the word type in the training corpus. Details of the breakdown of these features can be found in Section 3.3.

The acoustic-prosodic features s_i are learned via a CNN from energy (E) and pitch (f0) contours as described in (Tran et al., 2018) with a slight modification. In the original paper,

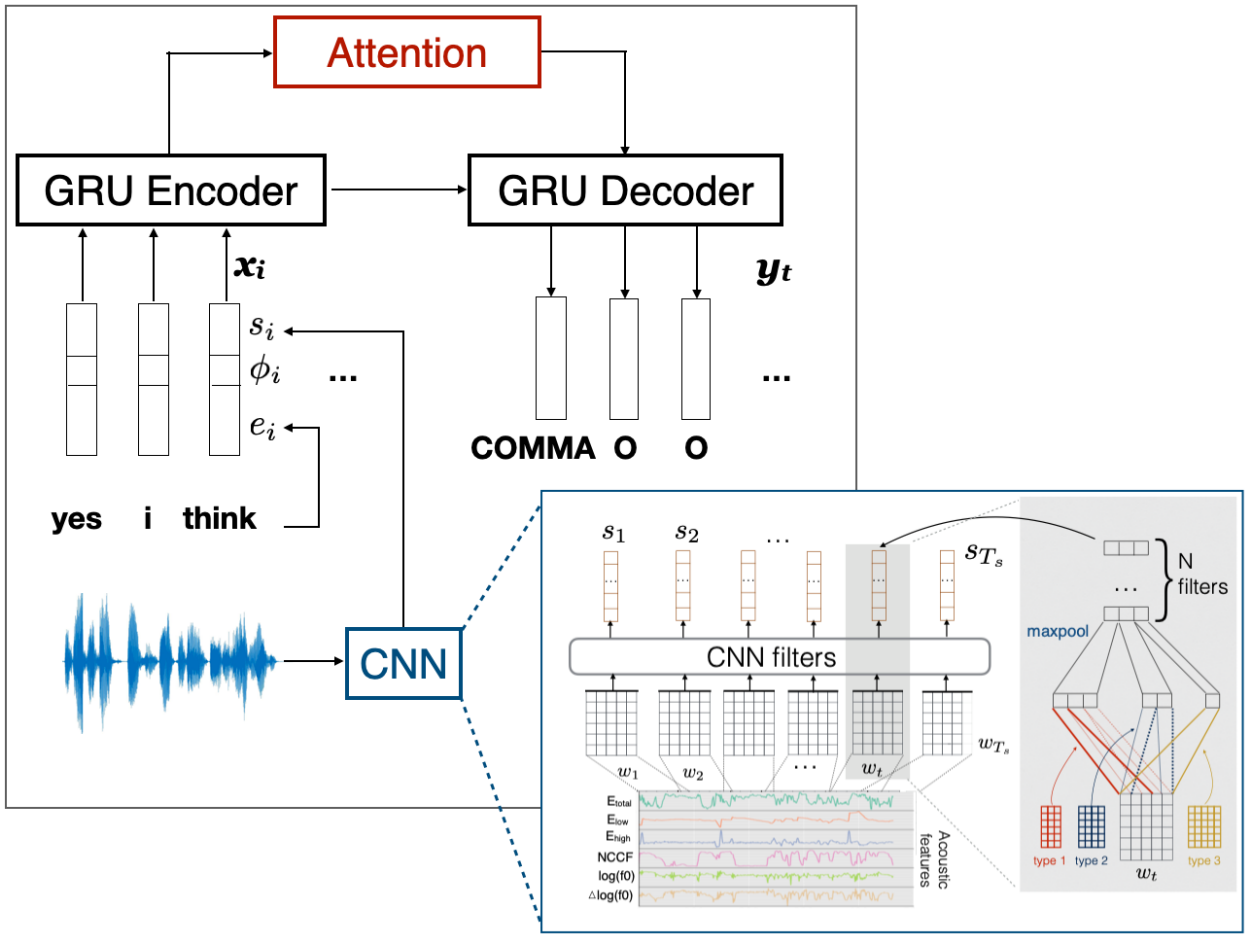


Figure 3.2: Schematic of the punctuation model. Each turn u is encoded via embeddings of the BERT-tokenized text, (optional) pause and duration embeddings, and (optional) convolved acoustic features.

$M/2$ frames to the left and $M/2$ frames to the right, from the center of the word, were convolved with the convolutional filters. In this work, the convolution window is shifted so that the center of the window is located at the *end* of words in order to capture the f0/E towards the end. This is motivated by the phenomenon where speakers change the pitch and/or energy of their voice at the end of the word to communicate a prosodic boundary. The impact of this shift is explored in Section 4.2.

The frame-level energy and pitch features corresponding to each word are then extracted based on word-level time alignments. Each sequence of f0/E frames corresponding to a time-aligned word (and potentially its surrounding context) is convolved with N filters of m sizes (a total of mN filters). The motivation for the multiple filter sizes is to enable the computation of features that capture information on different time scales. For each filter, we perform a 1-D convolution over the f0/E features with a stride of 1. Each filter output is max-pooled, resulting in mN-dimensional speech features s_i for word i. These prosody representations are jointly learned with the punctuation classification objective (cross-entropy).

In addition to a model trained on all features, we train a text-only model and a model where ϕ_i contains only the categorical pause feature. The best model uses 12-dimensional pause embeddings; the CNN has $N = 32$ sets of filters of widths $[5, 10, 25, 50]$, i.e. $m = 4$, totaling 128 filters. The context history chosen was $M = 1$ (meaning no history), and the RNN was the uni-directional GRU (Cho et al., 2014). Adam (Kingma and Ba, 2014) is used with initial learning rate 0.0001, halving when the performance tested on the validation set does not improve every 3 epochs.

3.5 Automatic Speech Recognizer

We use an off-the-shelf ASR system, ASpIRE (Povey et al., 2016), which was trained on Fisher conversational speech data (Cieri et al., 2004), available in Kaldi's (Povey et al., 2011) model suite. Briefly, the ASpIRE system was trained using a lattice-free maximum mutual information criterion, with computation efficiencies enabled by a phone-level language model and outputs at 1/3 the standard frame rate (one frame every 30 ms). The ASpIRE system

has a reported word error rate (WER) of 15.6% on the Hub5 ‘00 evaluation set. The WER on our SWBD data is 20.9% and 23.6% for the development and full test sets, respectively. Note that the ASpIRE ASR system is no longer available online.

Chapter 4
EXPERIMENTS

This chapter discusses different experimental setups in detail, presenting the results for each. The methods used in these experiments can be found in Chapter 3. The thesis research questions are answered in this chapter. The research questions to be addressed are as follows:

1. To what extent does prosody (beyond just pauses) improve punctuation prediction?

2. To what extent do ASR word errors impact the prediction performance?

3. How does explicit labeling of interruption points impact prediction results?

The first set of experiments address questions about the usefulness of prosody with the standard punctuation set used in most work {period (P.), question mark (Q?), comma (C,)}, augmented by a marker for an incomplete sentence (Inc-). A second set of experiments looks at incorporating the interruption point (IP+) as an additional category.

4.1 Experimental Setup

To train the model, we used the same optimizer, AdamW (Loshchilov and Hutter, 2019), with the same learning schedule as the provided implementation (Zhao and Kawahara, 2019). During training of the RNN, the weights of the transformer are fixed, instead of being updated with the RNN weights. The experiments in this work use the same model dimensions as the DA model in (Tran, 2020), exploring different CNN filter sizes and window location for feature extraction based on the development set results (see Section 4.2).

All models were trained with the same training set. However, the reported evaluation results slightly differ in what was included in the test set. Section 4.3 reports the results

from evaluating the model with the full test set, where all interruption points are mapped to no punctuation. Section 4.4 reports the results from evaluating the model with the IP test set, where 100% of the dialogues include disfluency annotations.

4.2 Prosody CNN Configuration

Before investigating the main questions, we conducted minor experiments to tweak hyper-parameters and determine the best feature configurations. First, we experimented with the convolutional filter sizes with the convolution window placed at the center of the word. The original sizes are {5, 10, 25, 50} frames and we tested {2, 5, 10, 15}, {10, 25, 50, 75}, {3, 9, 27, 81}. The variation in filters sizes is so that different time scales can be captured. Only the {10, 25, 50, 75} set showed slight improvement in the F1 score, but the numbers are not reported here because we decided that the improvement was insufficient to justify a change from the original model configurations in (Tran, 2020).

Secondly, we modified of the frame feature extraction process. In the original work, a fixed window of 100 frames (1 second) was centered at the middle of each word, which was then convolved with the CNN filters (Tran et al., 2019). We experimented with the placement of the convolution window in order to find out whether the location impacted the prediction performance when using the energy and pitch features. The prosodic cues relevant to punctuation prediction, specifically duration lengthening and phrase final tones, occur towards the end of the word. In addition, a pitch reset in the next word can signal the start of a new sentence. Therefore, we explored convolving frames centered at the end of the words. This placement will include pause frames and/or a portion of the next word as well.

Table 4.1 gives results for convolution window center placement. When the window is centered in the middle of each word, as in (Tran et al., 2018), the performance suffers by a small margin. Aligning the convolution window to the end of the word increases performance for period and question mark, but decreases performance for comma and incomplete mark. Based on the results, we decided to center the window at the end of words, which is the configuration used in the rest of the experiments and results in the following sections.

Table 4.1: F1 scores for prediction of expanded punctuation types with varying convolutional window center location for frame-based prosodic feature extraction.

location	C,	P.	Inc-	Q?	IP+	Macro F1
middle	.632	.762	.710	.696	.098	.579
end	.615	.781	.704	.710	.100	**.582**

4.3 Standard Punctuation Prediction Results

Tables 4.2 and 4.3 give the results for the 4-class, standard punctuation set. The results reported in this section are evaluations of the full test set as noted in Tables 3.1 and 3.2. Precision refers to how accurate the predicted results are, and recall refers to how many of the actual instances are correctly predicted by a model.

For hand transcripts (the top half of the tables), training the model with the f0 and energy features improved the performance (F1 score) of the model compared to only using the pause features. The greatest gain is for incomplete marks. Compared to the text only baseline, question marks deteriorate in performance when all prosody features are used whereas pauses help with question marks by a small amount. This result is surprising because we expected intonation provided by prosodic features to improve question detection. The macro F1 score improved by 1% relative to the all features result.

For the evaluation done on ASR transcripts, there is a similar trend in the results. Period and incomplete mark benefit from prosody where the full prosodic feature set gives a slight gain, but the main benefit is from pauses. Question mark improves with the pause only model, and there is no significant difference with all features. Comma does not improve in performance when prosody (for both pause and all features) is added. The main benefit from prosody is for questions and incomplete sentences. The macro F1 score improves by 3% relative to the all features result.

Based on both results, we can answer the first research question; prosody does help improve punctuation prediction. Commas are the most difficult to predict for both experiments, and they are likely to be more inconsistently annotated by human transcribers. As one would expect, general prediction performance degrades with the ASR transcripts for all classes: The 23.6% WER on this test set leads to a 17-18% reduction in macro F1. There is a particularly large drop for incomplete sentences, for both precision and recall. We hypothesized that the drop in performance on incomplete marks (roughly 35%) could be due to this class potentially encompassing more word errors than the other classes. Comparing the results of the text only model to the all features model, the prosody helps with punctuation prediction of ASR transcripts more than the manual transcripts.

Figure 4.1 displays the confusion matrices of various experiments, and the "O" class corresponds to words without punctuation. Figures 4.1a and 4.1b show that the prediction of all punctuation marks except question mark improved when using all prosody features for the hand transcript. For ASR transcripts, there is a gain on periods but a loss on commas (see Figures 4.1c and 4.1d). For both manual and ASR confusion matrices, we can observe that prosody reduces confusion of falsely predicted no punctuation labels.

4.4 Expanded Punctuation Set Results

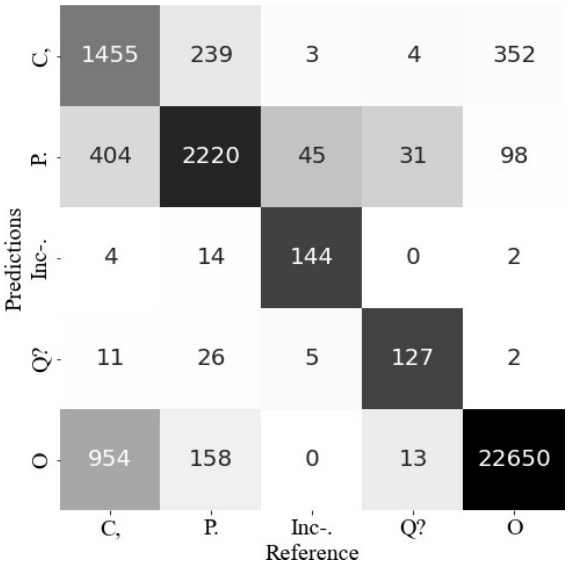
Since only 74% of the full test set contains disfluency annotations, a separate set of experiments was run on the subset of dialogues that contained disfluency annotations. This test set is referred to as the *IP test set*. All evaluation results in this section, including the 4-class experiments, are done with the IP test set. Tables 4.4 and 4.5 show the results for the 5-class punctuation set. The standard, 4-class punctuation experiments in the previous section had the interruption points unmarked. This meant that those were not included in the overall macro scores, so interruption detection that is reasonably high precision should not impact other classes much. This is confirmed for the manual transcripts. Using the pause feature alone degrades performance slightly on the hand transcriptions (from 0.761 to 0.759) and ASR transcriptions (from .600 to 0.595) when predicting with the 5-class punctuation set.

Table 4.2: F1 scores for prediction of *standard* punctuation types with different features. The results reported here are evaluations of the full test set as noted in Tables 3.1 and 3.2.

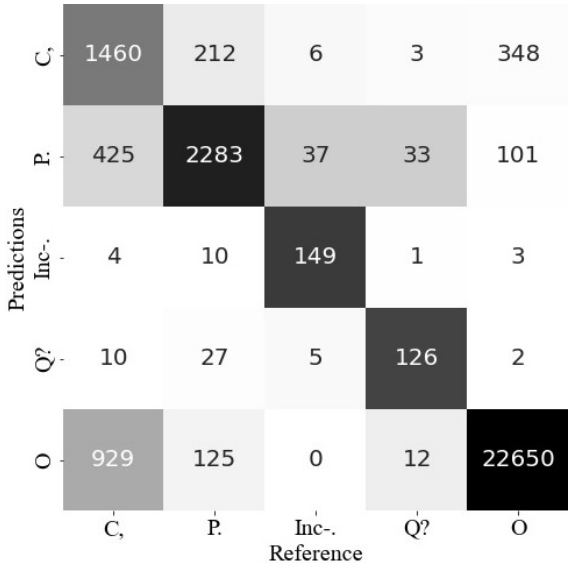
Manual	C,	P.	Inc-	Q?	Macro
text only	.596	.814	.798	.734	.736
pause only	.598	.819	.802	.735	.739
all features	.601	.825	.819	.730	.744
ASR	C,	P.	Inc-	Q?	Macro
text only	.533	.765	.495	.596	.597
pause only	.533	.774	.520	.623	.612
all features	.530	.777	.534	.617	.615

Table 4.3: Precision and recall, respectively, for prediction of *standard* punctuation types with different features. The results reported here are evaluations of the full test set as noted in Tables 3.1 and 3.2.

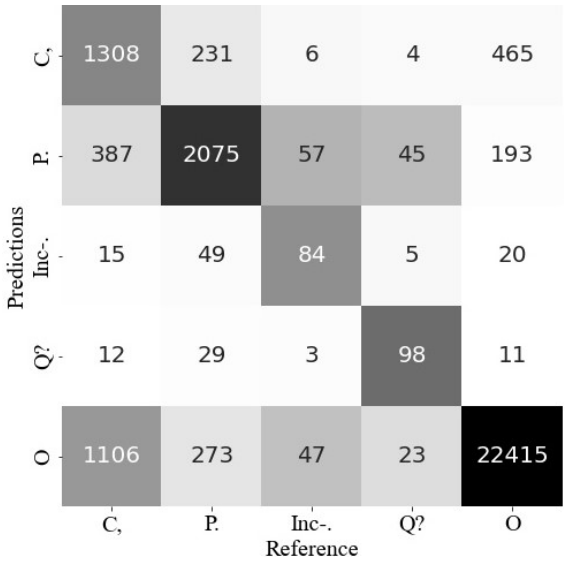
Manual	C,		P.		Inc-		Q?	
	P	R	P	R	P	R	P	R
text only	.71	.51	.79	.84	.88	.73	.74	.73
pause only	.72	.51	.79	.85	.89	.73	.76	.71
all features	.72	.52	.79	.86	.89	.76	.74	.72
ASR	C,		P.		Inc-		Q?	
	P	R	P	R	P	R	P	R
text only	.63	.46	.74	.79	.53	.47	.64	.56
pause only	.64	.46	.74	.81	.56	.49	.65	.59
all features	.63	.46	.74	.82	.58	.49	.67	.57



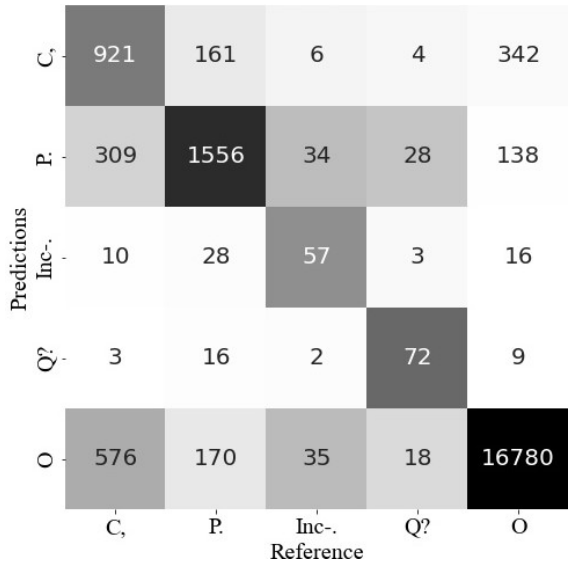
(a) Text only, Manual



(b) All features, Manual



(c) Text only, ASR



(d) All features, ASR

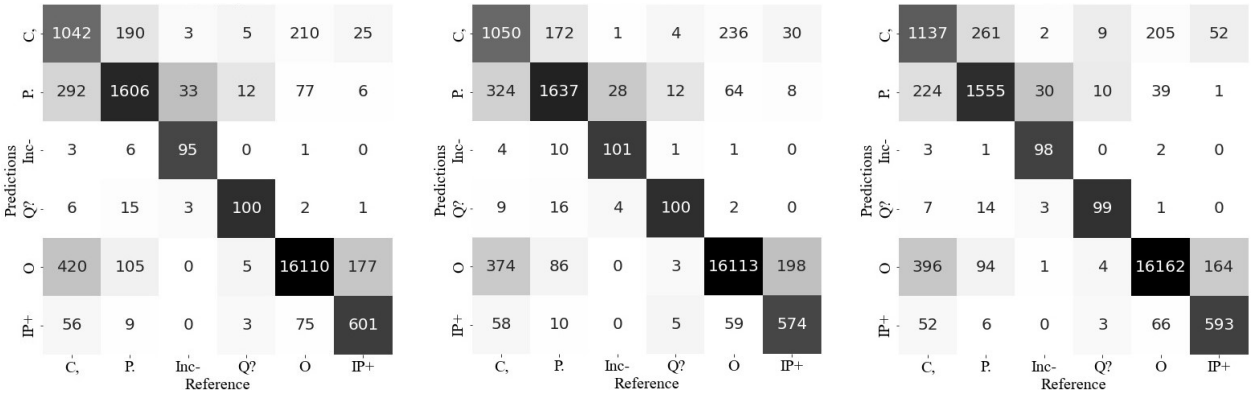
Figure 4.1: Confusion matrices of experiments with *standard* punctuation set. The matrices are reference-normalized, where the color of each cell reflects the relative weight of the given reference class.

The drop in the F1 score is because of question marks. Similar to the standard set results, prosody helps with prediction between text only and all features, with a relative increase of 1.6% for the macro F1 score. Also similar to the standard set, there is a 21% reduction in macro F1 associated with ASR transcripts Comparing the results of the text only model to the all features model, the improvement of hand transcript and ASR transcript evaluation is about the same (1.6% and 1.8% respectively).

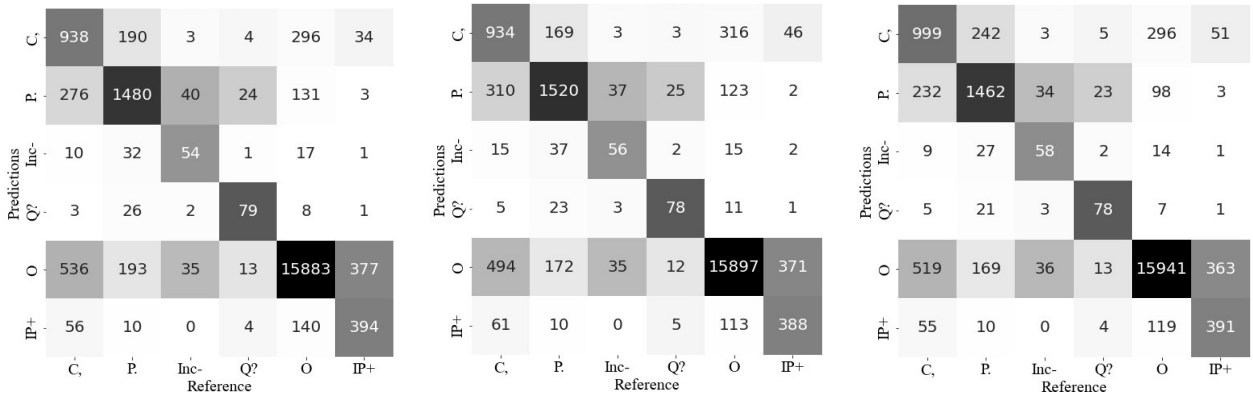
Figures 4.2a and 4.2c show that the prosody features helped reduce the confusion of commas the most. For periods and incomplete sentences, pauses are more helpful than using prosody. Figures 4.2d and 4.2f shows the impact of prosody by comparing its impact on the hand transcription and ASR transcription. The text only model has increased confusion across all labels in comparison to Figure 4.2a, especially high for incomplete marks and interruption points. Looking along the diagonal, it is apparent that trends seen in the manual hand transcript version is extended to the ASR results. Periods and interruption points are more confused as commas when prosody features are used.

Table 4.4: F1 scores for prediction of *expanded* punctuation types with different features. This result is for the IP test set, where all dialogues have disfluency annotations.

Manual	C,	P.	Inc-	Q?	IP+	Macro
text only	.633	.812	.795	.794	.773	.761
pause only	.634	.818	.805	.781	.757	.759
all features	.653	.821	.824	.795	.775	.773
ASR	C,	P.	Inc-	Q?	IP+	Macro
text only	.560	.760	.486	.648	.540	.600
pause only	.556	.766	.472	.632	.543	.595
all features	.572	.771	.516	.647	.547	.611



(a) Text only, Manual (b) Pause only, Manual (c) All features, Manual



(d) Text only, ASR (e) Pause only, ASR (f) All features, ASR

Figure 4.2: Confusion matrices of experiments with *expanded* punctuation set, evaluated on the full test set. The matrices are reference-normalized, where the color of each cell reflects the relative weight of the given reference class.

Table 4.5: Precision and recall, respectively, for prediction of *expanded* punctuation types with different features. This result is for the IP test set, where all dialogues have disfluency annotations.

Manual	C,		P.		Inc-		Q?		IP+	
	P	R	P	R	P	R	P	R	P	R
text only	.71	.57	.79	.83	.90	.71	.79	.80	.81	.74
pause only	.70	.58	.79	.85	.86	.75	.76	.80	.81	.71
all features	.68	.63	.84	.81	.94	.73	.80	.79	.82	.73
ASR	C,		P.		Inc-		Q?		IP+	
	P	R	P	R	P	R	P	R	P	R
text only	.61	.52	.75	.77	.52	.46	.66	.63	.59	.50
pause only	.61	.51	.74	.79	.48	.46	.64	.62	.61	.49
all features	.60	.55	.78	.76	.56	.48	.67	.62	.62	.49

4.5 Punctuation Set Analysis

In Table 4.3, the pause only model benefits prediction accuracy for ASR transcripts more than the manual transcripts. Prosody helped increase recall for periods in both transcript types. In Table 4.5, which is the 5-class punctuation set, the pause features mainly seem to change the location on the precision-recall curve, trading precision for recall, and vice versa.

One observation from looking at the macro F1 scores is that prosody substantially increases the performance prediction of incomplete marks in both hand and automatic transcripts. This could be due to the f0 and energy features that signal an incomplete boundary, which is different from the way a sentence would end for a period or a question mark. The improvement is more significant for ASR results. This outcome could suggest that the additional prosody features beyond pauses are likely compensating for the word errors that may occur near incomplete sentence boundaries. Of all punctuation labels, incomplete mark has a gain in precision consistently across all four categories recorded in the tables.

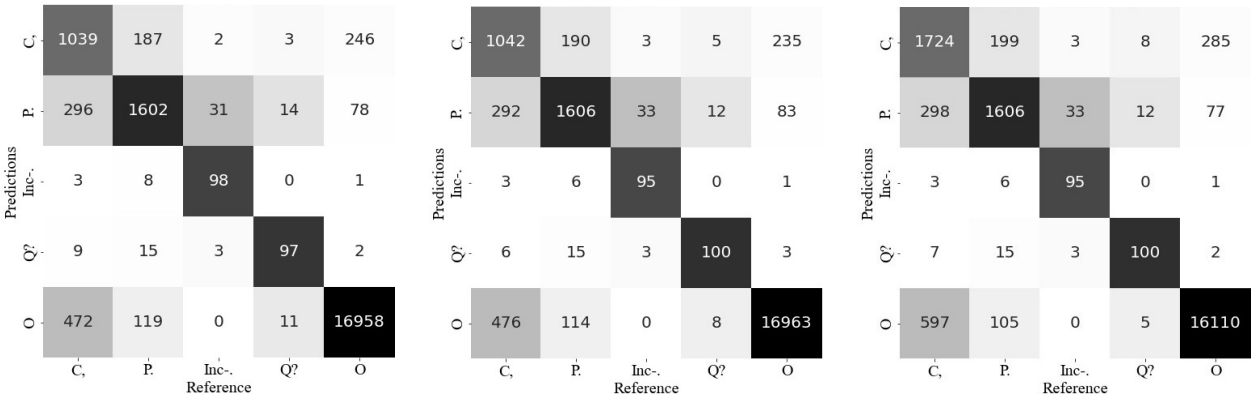
Surprisingly, question marks did not benefit from all prosody features as much as we expected. Acoustically, yes/no questions naturally end with a pitch raise which is easier to discern with prosody in addition to the word context alone. We hypothesized that without the pitch information, the accuracy would be lower. The lack of benefit could be due to the way we extracted the prosodic features, which may not have captured the parts of speech signals in a helpful manner. Overall, the prediction precision increases with prosody for all classes except for commas, as most punctuation marks have distinct acoustic characteristics.

In general, the computational costs of training a model using frame-based features (energy and pitch) are about 15 times as much as the token-based features (pause and word duration)—22 hours to 1.5 hours, respectively. However, the inference time is similar for all model configurations, so the gains make training with frame-based features worthwhile.

4.6 Analysis of Interruption Point

Interruption points are most often confused as no punctuation, followed by commas. This raised the question of whether results could be improved by explicitly modeling interruption points separately. We can observe how the explicit labeling of interruption points help or hinder the performance of other classes by mapping them to the no-punctuation category to simulate the 4-class evaluation and comparing this to the actual 4-class result. Figures 4.3g and 4.3h show a pair of confusion matrices that compare explicit modeling of interruption points to modeling punctuation without them. Figure 4.3h shows small improvements in prediction accuracy except for incomplete marks which is hurt by a small amount. Comparing the first two data columns of Table 4.6, we can see that separate modeling of interruption points yields a slight improvement when mapped to no punctuation. Therefore, explicitly labeling interruption points minimally contributes to the improvement in prediction performance.

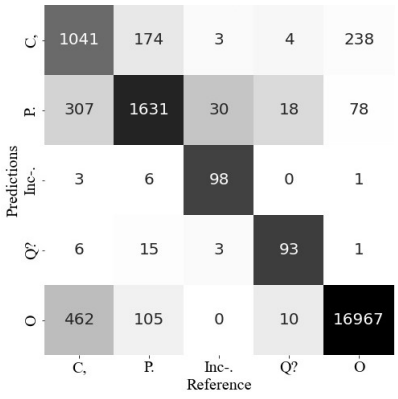
Note that in Figure 4.2c, `IP+` is secondmost confused as `C,` after `O`. Initially, we hypothesized that interruption points should be considered as no punctuation because these are inherently not associated with any punctuation marks. In comparing Figure 4.3h and 4.3i, and the last two data columns in Table 4.6, we find that interruption points may be better modeled as commas instead. However, keep in mind that the effect of merging the `IP+` to `C,` is much greater than merging to `O` because of the class size imbalance; no punctuation class is much larger. This raises a question: which other punctuation class could interruption points be merged with (besides no punctuation)? It is known that hand transcribers often had disagreements on what they labeled as incomplete sentences or interruption points because the nature of these structures could potentially align. A potential direction for future work could be to find out which other class that interruptions points might be best represented in. Though I only discussed the all features case, the finding is the same for text only and pause only models with the difference that the improvement is not as great as the all features case.



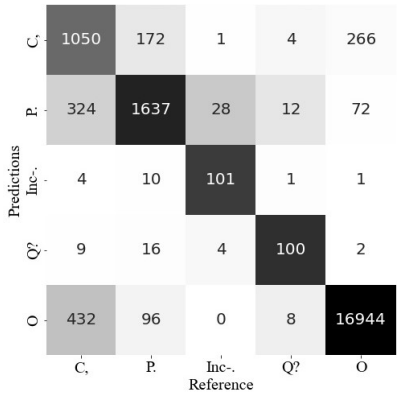
(a) Text only (standard)

(b) Text only, IP+ to O

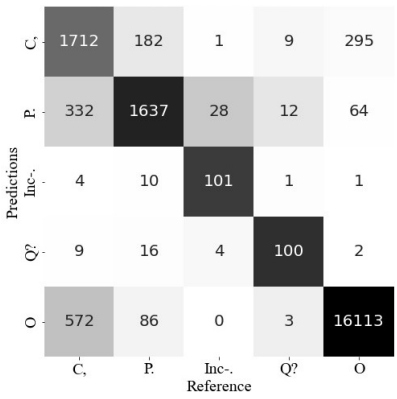
(c) Text only, IP+ to C,



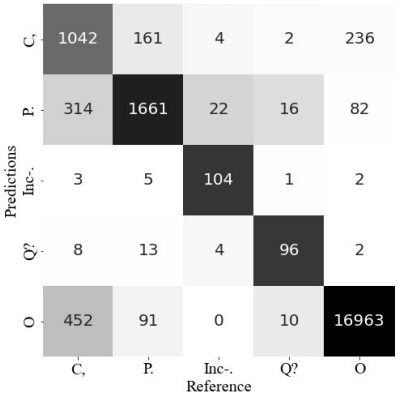
(d) Pause only (standard)



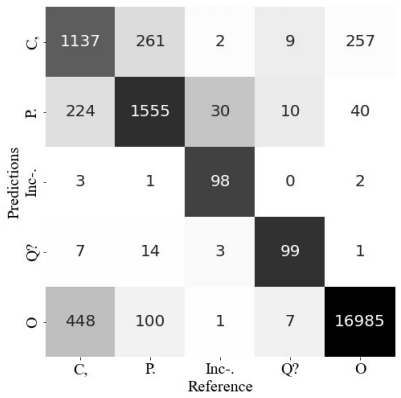
(e) Pause only, IP+ to O



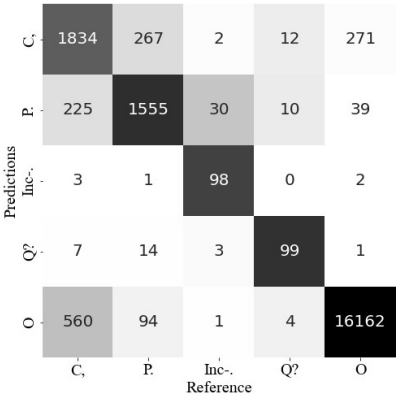
(f) Pause only, IP+ to C,



(g) All features (standard)



(h) All features, IP+ to O



(i) All features, IP+ to C,

Figure 4.3: Confusion matrices for comparing the effect of having interruption points. The 2nd column, `IP+ to O`, is the result for the expanded set result with the interruption points merged to no punctuation. The 3rd column, `IP+ to C,`, is the result for the expanded set result with the interruption points merged to commas.

Table 4.6: Macro F1 score of 4-class punctuation prediction given hand transcripts, comparing the 4-class model to the result of mapping the 5-class result to the 4 classes. All results reported in this table are evaluated on the IP test set, where all dialogues have disfluency annotations.

	4-class (IP test)	5:4-class (O)	5:4-class (COMMA)
text only	.754	.758	.778
pause only	.757	.759	.778
all features	.768	.773	.793

Chapter 5

CONCLUSIONS

5.1 Summary

There is no doubt that prosody helps people comprehend sentences as they listen in a conversation. One aspect is the cues associated with phrase boundaries that would be marked by punctuation in written text especically for the incomplete marks. To summarize the research findings, prediction performance benefits from prosody, and the impact is greater for ASR transcribed text. As expected, pause features help with punctuation prediction, but using all prosody features (pauses, word duration, pitch, and energy) is more helpful than using pauses alone. For disfluencies, marking the interruption points as its own punctuation helped in overall prediction performance. Although, it may be more appropriate to consider interruption points as commas rather than no punctuation. Separately classifying interruption points may be helpful for scenarios where it is useful to clean up disfluencies, such as real time captioning where repeated words and other disfluencies should be filtered. Lastly, the use of durational and acoustic features adds significant computational cost in training, but does benefit the overall model performance. The long training may be worthwhile as the difference in inference time is negligible.

It is important to consider the results of this work in relation to other punctuation prediction studies but the difference in model architecture and speaking styles of data make it difficult to compare the results directly. Here I highlight the results of a few studies in order to put this work in context. Makhija et al. (2019) use BertPunc, a model combining BERT and an LSTM, to train a punctuation prediction model with word embeddings only. The relevant data is TED talk transcriptions which is different from conversational speech. Their best model had an overall F1 score of 0.814. Lin and Wang (2020) use a joint approach

of combining punctuation prediction and disfluency detection on the SWBD dataset. Their best model, a combination of those two tasks and predicting disfluency after punctuation, yields an overall macro F1 score of 0.803. This is model is trained with word embeddings only. The highlighted studies report the scores on the evaluation of hand transcripts. Both studies have a higher overall macro F1 score than the results presented in this work, but the results would be roughly comparable or better, because those studies may have included commas that are associated with filled pauses. Such commas are considered to be freebies because of how easy it is to correctly predict a comma for a filled pause based on the text.

In recent studies, the improvements in performance have been small. As it has been difficult to achieve large gains over the punctuation prediction task, the question regarding its difficulty rises. One potential explanation for this could be due to the low agreement in human transcriptions. If two transcribers were asked to label the same excerpt, the result would be different from one another. The lack of agreement in punctuation markings may be the upper bound of how well punctuation models can perform at this time. There are no studies assessing the extent to which people disagree on the use of punctuation (e.g., commas) for spontaneous speech so the upper bound is unknown.

5.2 Future Work

There remain several aspects of this work to be further investigated. It would be good to check the WER of the IP test set, and also look into the WER of each punctuation category which would help in the analysis of how each class impacts the performance. Interruption points appear to help most as commas, and commas were often used at IPs in the SWBD transcripts. However, the class imbalance may have yielded misleading results. It would be interesting to look at the impact of merging IPs into other punctuation as well.

Lastly, the method of extracting pause and word duration features may need to be revised to reflect characteristics of more recent ASR systems. The impact of prosody could change with more accurate ASR but less accurate times. The future direction of this work could be towards exploring different ways to represent prosody features. The work in this thesis utilizes

a preexisting model used for joint dialogue act segmentation and classification tasks with minor modifications in feature extraction. Instead of using Kaldi's pitch extraction method which localizes on the first harmonic, one could explore ways to capture all harmonics so that the features are enriched. A spectrogram could be integrated into the current set up and take advantage of the CNN with the fact that a spectrogram is similar to an image.

Another option to be explored is to use a larger punctuation set which includes other punctuation such as exclamation marks, ellipsis, semicolon, etc. Depending on the speaking style, many punctuation marks may not be as applicable. This work explores spontaneous conversational speech, but other styles, such as story telling, could be interesting to study.

BIBLIOGRAPHY

Disfluency. URL https://www.oxfordbibliographies.com/view/document/obo-9780199772810/obo-9

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *Proc. ICLR*, 2015.

D. Beeferman, A. Berger, and J. Lafferty. Cyberpunc: A lightweight punctuation annotation system for speech. In *Proc. ICASSP*, page 689–692, 1998.

K. Cho, B. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL https://www.aclweb.org/anthology/D14-1179.

H. Christensen, Y. Gotoh, and S. Renals. Punctuation annotation using statistical prosody models. In *Proc. ISCA Workshop on Prosody in Speech Recognition and Understanding*, 2001.

C. Cieri, D. Graff, O. Kimball, D. Miller, and K. Walker. Fisher English training speech part 1 transcripts, ldc2004t19, 2004.

H. H. Clark and J. E. Fox Tree. Using uh and um in spontaneous speaking. *Cognition*, 84 (1):73–111, 2002. ISSN 0010-0277. doi: https://doi.org/10.1016/S0010-0277(02)00017-3. URL https://www.sciencedirect.com/science/article/pii/S0010027702000173.

V.-T. Dang, T. Zhao, S. Ueno, H. Inaguma, and T. Kawahara. End-to-End Speech-to-Dialog-Act Recognition. In *Proc. Interspeech*, pages 3910–3914, 2020.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL*, pages 4171–4186, 2019.

M. Fang, H. Zhao, X. Song, X. Wang, and S. Huang. Using bidirectional LSTM with BERT for Chinese punctuation prediction. In *Proc. IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*, pages 1–5, 2019.

J. J. Godfrey and E. Holliman. *Switchboard-1 Release 2*. Linguistic Data Consortium, 1993.

A. Gravano, M. Jansche, and M. Bacchiani. Restoring punctuation and capitalization in transcribed speech. In *Proc. ICASSP*, page 4741–4744, 2009.

Y. Guo, H. Wang, and J. Van Genabith. A linguistically inspired statistical model for Chinese punctuation generation. *ACM Transactions on Asian Language Information Processing (TALIP)*, 9(2):1–27, 2010.

X. He, Q. Tran, W. Havard, L. Besacier, I. Zukerman, and G. Haffari. Exploring Textual and Speech information in Dialogue Act Classification with Speaker Domain Adaptation. In *Proc. Australasian Language Technology Association Workshop*, pages 61–65, Dunedin, New Zealand, December 2018. URL `https://www.aclweb.org/anthology/U18-1007`.

J. Hirschberg and C. Nakatani. Acoustic Indicators of Topic segmentation. In *Proc. International Conference on Spoken Language Processing (ICSLP)*, 1998.

J. Hirschberg, D. Litman, and M. Swerts. Prosodic and Other Cues to Speech Recognition Failures. *Speech Communication*, 43:155–175, 06 2004. doi: 10.1016/j.specom.2004.01.006.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

J. Huang and G. Zweig. Maximum entropy model for punctuation annotation from speech. In *Proc. ICSLP*, 2002.

D. Jurafsky, E. Shriberg, and D. Biasca. Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual, Draft 13. Technical Report 97-02, University of Colorado, Boulder Institute of Cognitive Science, Boulder, CO, 1997.

J. G. Kahn and M. Ostendorf. Joint reranking of parsing and word recognition with automatic segmentation. *Computer Speech & Language*, 26(1):1–51, 2012.

J. G. Kahn, M. Lease, E. Charniak, M. Johnson, and M. Ostendorf. Effective Use of Prosody in Parsing Conversational Speech. In *Proc. HLT/EMNLP*, 2005.

J.-H. Kim and P. Woodland. A Combined Punctuation Generation and Speech Recognition System and Its Performance Enhancement Using Prosody. *Speech Communication*, 41 (4):563–577, 2003. ISSN 0167-6393. doi: https://doi.org/10.1016/S0167-6393(03)00049-9. URL `http://www.sciencedirect.com/science/article/pii/S0167639303000499`.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980, 2014.

O. Klejch, P. Bell, and S. Renals. Sequence-to-sequence models for punctuated transcription combining lexical and acoustic features. In *Proc. ICASSP*, pages 5700–5704, 2017. doi: 10.1109/ICASSP.2017.7953248.

J. Kolář, E. Shriberg, and Y. Liu. Using Prosody for Automatic Sentence Segmentation of Multi-party Meetings. In *Text, Speech and Dialogue*, pages 629–636, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

T. Levy, V. Silber-Varod, and A. Moyal. The effect of pitch, intensity and pause duration in punctuation detection. In *Proc. IEEE Convention of Electrical and Electronics Engineers in Israel*, pages 1–4, 2012.

R. J. Lickley. *Fluency and Disfluency*, chapter 20, pages 445–474. John Wiley Sons, Ltd, 2015. ISBN 9781118584156. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118584156.ch20.

B. Lin and L. Wang. Joint prediction of punctuation and disfluency in speech transcripts. In H. Meng, B. Xu, and T. F. Zheng, editors, *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 716–720. ISCA, 2020. doi: 10.21437/Interspeech.2020-1277. URL https://doi.org/10.21437/Interspeech.2020-1277.

Y. Liu, A. Stolcke, E. Shriberg, and M. Harper. Comparing and Combining Generative and Posterior Probability Models: Some Advances in Sentence Boundary Detection in Speech. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 64–71, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W04-3209.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

K. Makhija, T.-N. Ho, and E.-S. Chng. Transfer learning for punctuation prediction. In *Proc. APSIPA Annual Summit and Conference*, pages 268–273, 2019.

A. Margolis, K. Livescu, and M. Ostendorf. Domain adaptation with unlabeled data for dialog act tagging. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 45–52, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL https://aclanthology.org/W10-2607.

A. Moró and G. Szaszák. A prosody inspired RNN approach for punctuation of machine produced speech transcripts to improve human readability. In *Proc. IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pages 219–224, 2017.

R. Pappagari, P. Zelasko, A. Mikolajczyk, P. Pezik, and N. Dehak. Joint prediction of truecasing and punctuation for conversational speech in low-resource scenarios. In *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1185–1191, 2021.

D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý. The Kaldi Speech Recognition Toolkit. In *Proc. ASRU*, 2011.

D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur. Purely sequence-trained neural networks for asr based on lattice-free mmi. In *Proc. Interspeech*, 2016.

E. Shriberg, A. Stolcke, D. Jurafsky, N. Coccaro, M. Meteer, R. Bates, P. Taylor, K. Ries, R. Martin, and C. Ess-Dykema. Can Prosody Aid the Automatic Classification of Dialog Acts in Conversational Speech? *Language and Speech*, 41(3-4):443–492, 1998.

K. Silverman, M. Beckman, J. Pitrelli, M. Ostendorf, C. Wightman, P. Price, J. Pierrehumbert, and J. Hirschberg. ToBI: A Standard for Labeling English Prosody. In *Proc. ICSLP*, 1992.

A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. Van, and M. Meteer. Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics (COLING)*, 26(3):339–374, 2000. URL `https://www.aclweb.org/anthology/J00-3003`.

M. Sunkara, S. Ronanki, K. Dixit, S. Bodapati, and K. Kirchhoff. Robust prediction of punctuation and truecasing for medical asr. In *Proc. Workshop on NLP for Medical Conversations*, pages 53–62, 2020.

T. Tran. *Neural Models for Integrating Prosody in Spoken Language Understanding*. PhD thesis, University of Washington, 2020.

T. Tran and M. Ostendorf. Assessing the use of prosody in constituency parsing of imperfect transcripts. In *Proc. Interspeech*, pages 2626–2630, 2021.

T. Tran, S. Toshniwal, M. Bansal, K. Gimpel, K. Livescu, and M. Ostendorf. Parsing speech: a neural approach to integrating lexical and acoustic-prosodic information. In *Proc. NAACL*, pages 69–81, June 2018. doi: 10.18653/v1/N18-1007. URL https://www.aclweb.org/anthology/N18-1007.

T. Tran, J. Yuan, Y. Liu, and M. Ostendorf. On the Role of Style in Parsing Speech with Neural Models. In *Proc. Interspeech*, pages 4190–4194, 2019. doi: 10.21437/Interspeech.2019-3122. URL http://dx.doi.org/10.21437/Interspeech.2019-3122.

G. Tür, D. Hakkani-Tür, A. Stolcke, and E. Shriberg. Integrating Prosodic and Lexical Cues for Automatic Topic Segmentation. *Computational Linguistics*, 27:31–57, 2001.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.

M. Wagner and D. G. Watson. Experimental and theoretical advances in prosody: A review. *Language and cognitive processes*, 25:905–945, 2010.

M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks, 2013. URL https://arxiv.org/abs/1311.2901.

P. Zelasko, P. Szymanski, J. M. A. Szymczak, Y. Carmiel, and N. Dehak. Punctuation prediction model for conversational speech. In *Proc. Interspeech*, pages 2633–2637, 2018.

T. Zhao and T. Kawahara. Joint Dialog Act Segmentation and Recognition in Human Conversations Using Attention to Dialog Context. *Computer Speech & Language*, 57: 108–127, 2019. ISSN 0885-2308.

Y. Zhao, C. Wang, and G. Fu. A CRF sequence labeling approach to Chinese punctuation prediction. In *Proc. Pacific Asia Conference on Language, Information, and Computation*, pages 508–514, 2012.