Towards End-2-end Learning for Predicting Behavior Codes from Spoken Utterances in Psychotherapy Conversations

Karan Singla[1], Zhuohao Chen[1], David C. Atkins[2], Shrikanth Narayanan[1]
[1]University of Southern California, Los Angeles, USA
[2]University of Washington, Seattle, WA, USA
{singlak, zhuohaoc}@usc.edu, datkins@uw.edu, shri@sipi.usc.edu

Abstract

Spoken language understanding tasks usually rely on pipelines involving complex processing blocks such as voice activity detection, speaker diarization and Automatic speech recognition (ASR). We propose a novel framework for predicting utterance level labels directly from speech features, thus removing the dependency on first generating transcripts, and transcription free behavioral coding. Our classifier uses a pretrained Speech-2-Vector encoder as bottleneck to generate word-level representations from speech features. This pretrained encoder learns to encode speech features for a word using an objective similar to Word2Vec. Our proposed approach just uses speech features and word segmentation information for predicting spoken utterance-level target labels. We show that our model achieves competitive results to other state-of-the-art approaches which use transcribed text for the task of predicting psychotherapy-relevant behavior codes.

1 Introduction

Speech interfaces have seen a widely growing trend and this has brought about increasing interest in advancing computational approaches to spoken language understanding (SLU). (Tur and De Mori, 2011; Xu and Sarikaya, 2014; Yao et al., 2013; Ravuri and Stolcke, 2015). SLU systems often rely on Automatic speech recognition (ASR) for generating lexical features. The ASR output is then used for the target natural language understanding task. Furthermore, end-2-end SLU systems for various applications, including speech synthesis (Oord et al., 2016), ASR tasks (Amodei et al., 2016; Chan et al., 2016; Soltau et al., 2016) and speech-2-text translation (Chung et al., 2019) have shown promising results. Recently (Haque et al., 2019) propose a method for learning audio-linguistuc embedding but that too depends on using transcribed text.

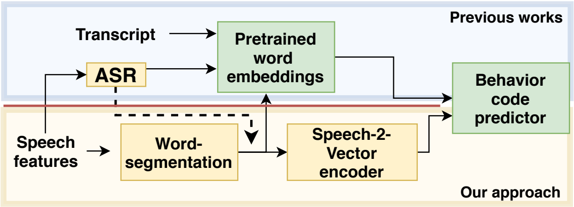

Figure 1: Upper part describes most of existing approaches which either use ASR or manual transcripts. Lower part shows our proposed approach where we predict behavior codes without using transcripts

Due to the nature of the speech processing pipeline, natural language understanding tasks suffer from two major problems, 1) error propagation through ASR leading to noisy lexical features 2) loss of rich information which supplement lexical features, such as prosodic and acoustic expressive speech patterns.

In this paper, we propose a framework to address the problem of predicting behavior codes directly from speech utterances. We focus on data from Motivational Interviewing (MI) sessions, a type of talk-based psychotherapy focused on behavior change. In psychology research and clinical practice, behavioral coding is often used to understand process mechanisms and therapy efficacy and outcomes. Behavior codes are annotated by an expert at an utterance level (or interaction level) by listening to the session. Examples of utterance level behavior codes include if there was a simple of complex reflection by the therapist of their patient's previous utterance(s). Several approaches have been proposed for automatic prediction of behavior codes, mainly using lexical features and/or linguistic features such as information from dependency trees (Xiao et al., 2016; Tanana et al., 2016; Pérez-Rosas et al., 2017; Cao et al., 2019; Gibson et al., 2019). Recent works (Singla et al., 2018; Chen et al., 2019) reveal that using acoustic and

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3797–3803
July 5 - 10, 2020. ©2020 Association for Computational Linguistics

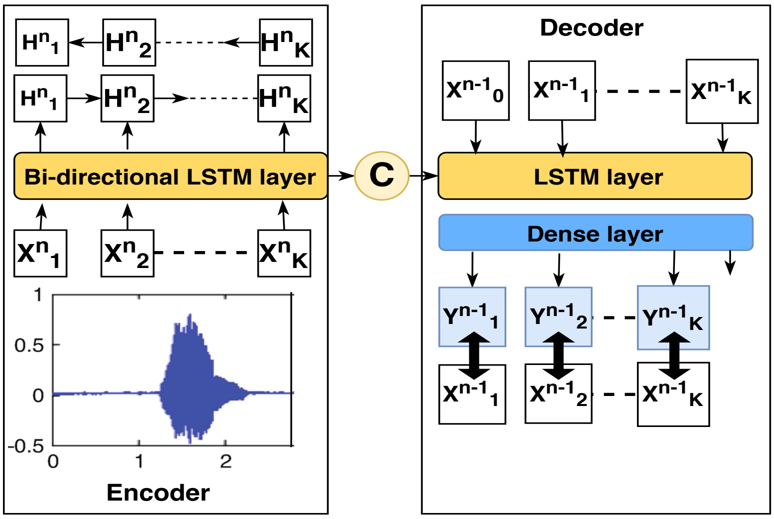

Figure 2: Speech signal to word encoder (SSWE) which uses sequence-2-sequence framework for generating representations of context words given a word.

prosodic features in addition to lexical features outperforms single modality models.

Speech2Vec (Chung and Glass, 2018) has shown that high quality word representations can be learnt by just using speech features. It learns word representations in an unsupervised manner using an objective similar to the Skipgram objective of Word2Vec (Mikolov et al., 2013) (a word representation should be representative of its context words) and sequence-to-sequence framework. However, Speech2Vec only aims to learn word representations which are averaged spoken-word representations of that word in the corpus. Our proposed approach aims to exploit speech signal to word encoder learnt using an architecture similar to Speech2Vec as lower level dynamic word representations for the utterance classifier. Thus, our system never actually needs to know what word it is but only word segmentation information. We hypothesize that word segmentation information can be obtained with cheaper tools, e.g. a supervised word segmentation system (Tsiartas et al., 2009) or a heuristics based system based on acoustic and prosodic cues (Junqua et al., 1994; Iwano and Hirose, 1999). We plan to investigate the effect of noise in word boundaries on encoder quality in the future.

Our end-2-end transcription-free approach is similar and perhaps even motivated some of the previous works. There have been some works (Serdyuk et al., 2018; Lugosch et al., 2019) which perform prediction tasks directly from speech signals but lack in capturing the underlying linguistic structure of a language (sentences break into words for semantics). We believe capturing some of the important linguistic units (e.g. words) are important for spoken language understanding. (Qian et al., 2017) is most similar to our work in terms of overall architecture as they also first get word level representations and then use the encoder for utterance level prediction. However (Qian et al., 2017) uses transcribed word transcriptions but we only use word boundaries for ASR-free end-2-end spoken language understanding. As shown in Figure 1, most previous works follow the upper pipeline. They start with a transcript (manually generated or through an ASR), which is first segmented into utterances. They then use word-embeddings for each word in the transcript before feeding it into a classifier to predict target behavior codes.

Our approach shows competitive results when compared to state-of-the-art models which use transcribed text. Our target application domain in this work is psychotherapy. While utterance level behavior coding is a valuable resource for psychotherapy process research, it is also a labor intensive task for manual annotation. Our proposed method which does not rely on transcripts should help with cheaper and faster behavioral annotation. We believe this framework can be a promising direction to directly perform classification tasks given a spoken utterance.

2 Our Approach

We first learn a word-level speech signal to word encoder using a sequence-to-sequence framework.

Speech-2-Vector follows the learning objective similar to Skipgram architecture of Word2Vec. We then use the pre-trained encoder to predict behavior codes.

2.1 Speech signal to word encoder

Our Speech signal to word encoder (SSWE) encoder is an adaptation of Speech2Vec (Chung and Glass, 2018) which in turn is motivated by Word2Vec's skipgram architecture. The model learns to predict context words given a word. But unlike Word2Vec, in SSWE, each word is represented by a sequence of speech frames. We adopt the widely known sequence-to-sequence architecture to generate context words given a spoken word. Our model generates speech features for context words $(X_{n-4}, X_{n-3}, ..., X_{n+4})$ given speech features for a word X_n. As input for word X_n, it takes $K * 13$ dimensional MFCC features extracted from every 25 ms window of speech audio using a frame rate of 10ms. K is the maximum number of frames a spoken word can have. This input is then processed through a bidirectional LSTM layer (Hochreiter and Schmidhuber, 1997) to generate the context vector C. C is then used by a unidirectional LSTM decoder to generate the speech features for words in context $(Y_{n-4}, Y_{n-3}, ..., Y_{n+4})$. We optimize the model by minimizing the mean squared loss between predicted and target outputs: $\sum_{i=1}^{k} \left\|X^i - Y^i\right\|^2$. Following this approach, our system never uses any form of explicit transcriptions for learning the encoder, just only the word boundaries. Figure 2 gives a pictorial description of this process.

Our Speech-2-Vector encoder is trained using a speech corpus and word segmentation information. In our setup, we assume we have high quality word segmentation information. For the purpose of our experiments, we obtain the word segmentation information using a Forced-aligner (Ochshorn and Hawkins, 2016) (it uses transcripts but we only use it for word segmentation, we plan to replace it with other tool). The forced aligner primarily gives boundaries for the start and end of a word, which are then used to get speech features for a word. We hypothesize that learning word segmentation is a cheaper task than training a full-blown ASR.

2.2 Utterance classifier

Figure 3 shows the picturesque view of our utterance classifier. Given a word-segmented utterance, we first process speech features for each word to get word-level representations (W_i..... W_n). We then learn a function c = f(W) that maps W to a behavioral code $c1, 2, ..., C$, with C being the number of defined target code types.

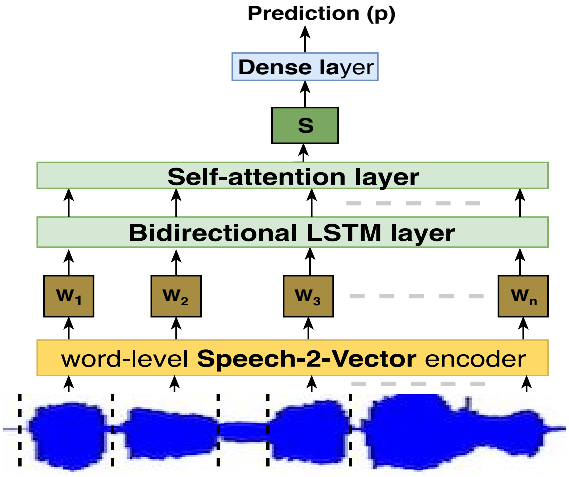

Figure 3: Classifier to predict behavior codes which takes input a word segmented speech signal and also uses pretrained Speech-2-Vector encoder to get word level representations.

Code	Description	#Train	#Test
FA	Facilitate	1194	496
GI	Giving information	12241	4643
RES	Simple reflection	4594	1902
REC	Complex reflection	3613	1235
QUC	Closed question (Yes/No)	4393	2066
QUO	Open question (Wh-type)	3871	1445
MIA	MI adherent	2948	1521
MIN	MI non-adherent	890	433
Total		33744	13741

Table 1: Data statistics for Behavior code prediction in Motivational Interviewing Psychotherapy

We use a parametric composition model to construct utterance-level embeddings from word-level embeddings. Word-level representations $(W_i, ..., W_n)$ are then fed into a bidirectional LSTM layer to contextualize the word embeddings. Contextualized word embeddings are then fed to a self-attention layer to get a sentence representation S which is then used to predict the behavior code for an utterance using a dense layer which projects it to C dimensions using a softmax operation. We use a self-attention mechanism similar to the one proposed in (Yang et al., 2016)

3 Dataset

We experiment with two datasets for training the S2V encoder: first on the LibreSpeech Corpus

(Panayotov et al., 2015) (500 hour subset of broadband speech produced by 1,252 speakers) and second, directly on our classifier training data, which we describe below.

For classification, we use data from Motivational Interviewing sessions (a type of talk based psychotherapy) for addiction treatment presented in (Tanana et al., 2016; Pérez-Rosas et al., 2017). There are 337 transcribed sessions (approx. 160 hours of audio) coded by experts at the utterance level with behavioral labels following the Motivational Interviewing Skill Code (MISC) manual (Miller et al., 2003). Each human coder segmented talk turns into utterances (i.e., complete thoughts) and assigned one code per utterance for all utterances in a session. The majority of sessions were coded once by one of three expert coders.

In this paper, we use the strategy proposed by (Xiao et al., 2016) grouping all counselor codes into 8 categories (described in Table 1). We remove backchannels without timestamps which cannot be aligned and split the data into training and testing sets by sessions with roughly 2:1 ratio. This split is consistent with all compared works.

4 Training details

Speech-2-Vector Encoder: We implemented the model with PyTorch (Paszke et al., 2017). Similar to (Chung and Glass, 2018), we also adopted the attention mechanism which enables the Decoder to condition every decoding step on the last hidden state of the Encoder (Subramanian et al., 2018). The window size was set to 4. We train the model using stochastic gradient descent (SGD) with learning rate of $1e*-3$ and batch size of 64 (spoken-word, context) pairs. We experimented with hyperparameter combinations for: using bidirectional or unidirectional RNNs, using GRU vs LSTM cell, number of LSTM hidden layers and learning rates. We found there was not a big difference in encoder output quality with higher dimensions. Therefore, we use a 50 dimensional LSTM cell, thus the resulting encoder output becomes 100 (Bidirectional last hidden states) + 100 (cell state) = 200 dimensions.

Utterance Classifier: The chosen batch size was 40 utterances. The LSTM hidden state dimension is 50. We use dropout at the embedding layer with drop probability 0.3. The dense layer is of 100 dimensions. The model is trained using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001 and an exponential decay of 0.98 after 10K steps (1 step = 40 utterances). Similar to prior work, we also weight each sample according to normalized inverse frequency ratio.

Model	**Word embeddings Data**	**F1-score**
Word2Vec[†]	Google-wiki	0.53
Word2Vec[†]	Indomain	0.56
Speech2Vec[†]	LibreSpeech	0.58
Speech2Vec[†]	Libre+Indomain*	**0.60**

Table 2: Using word embeddings learnt using speech features (Speech2vec) vs Word2Vec. * marks that model was only fine tuned for in-domain data. † marks that all these classifiers were trained end-2-end

5 Experiments & Results

Speech2Vec vs Word2Vec: Table 2 shows results where we compare performance of the system when we use lexically-derived word embeddings (word2Vec) vs speech-features derived word embeddings (Speech2Vec). If a word appears in a corpus n times, then speech2vec uses a system similar to our Speech-2-Vector encoder and averages them to get a word embedding for that dictionary word. Results confirm two main observations: 1) It is better to learn/fine-tune the word embeddings on an in-domain dataset. 2) Speech2Vec that learns word embeddings based on different spoken variations of word provides better results for behavior code prediction. This result is consistent with findings from (Singla et al., 2018; Chen et al., 2019) where it is shown that acoustic-prosodic information can provide complementary information for predicting behavior codes and hence, produce better results. One challenge is that SSWE and Speech2Vec generally needs large amount of transcribed data to learn high quality word embeddings. Therefore, we first train SSWE on a general speech corpus (here, LibreSpeech (Libre)) before fine-tuning it on our classifier training data (results with $*$ show this experiment).

Transcriptions vs. No Transcriptions: Methods discussed above still rely on transcriptions to know what the word is. However, our proposed method does not use any explicit transcription but only the word segmentation information. Results in Table 3 show that using a pre-trained Speech-2-Vector encoder as a building block to get word representations can lead to competitive results to other methods which rely heavily on first generating transcripts of the spoken utterance. Here

Model	Pretrain data	F1-score
Majority class	-	0.33
Single-modality		
Word2Vec†	Indomain	0.56
Prosodic	Indomain	0.42
Multimodal		
Word2Vec+Prosodic†	Indomain	0.58
Speech2Vec†	Libre+Indomain*	**0.60**
Speech-only (Our approach)		
SSWE	Indomain	0.49
SSWE†	Indomain	0.44
SSWE	Libre+Indomain*	**0.56**
SSWE†	Libre+Indomain*	0.50

Table 3: We compare our proposed approach to previous approaches. Results in red are for the systems that do not use any transcriptions, only word segmentation information.

we also compare our model to the multimodal approach proposed by (Singla et al., 2018; Chen et al., 2019) where they use word-level prosodic features along with lexical word embeddings. *Prosodic* and *Word2Vec+Prosodic*† show results for this system.

Table 3 also shows that doing end-2-end training (results with *) where our Speech-2-Vector encoder is also updated by the classifier loss generates poor results. We hypothesize that it can be due to the fact that our behavior code prediction data was split to minimize the speaker overlap. Thus it becomes easier to overfit when we fine-tune it on some speaker-related properties instead of generalizing for behaviour code prediction task.

6 Conclusions

We show that comparable results can be achieved for behavior code prediction by just using speech features and without any ASR or human transcriptions. Our approach still depends on word segmentation information, however, we believe obtaining word segmentation from speech is comparatively easier than building a high quality ASR. The evaluation results show the application significance of an end-2-end speech to behavioral coding for psychotherapy conversations. This allows for building systems that do not include explicit transcriptions, an attractive option for privacy reasons, when the end goal (as determined by the behavioral codes) is to characterize the overall quality of the clinical encounter for training or quality assurance.

7 Future work

The results still vary and are worse compared to using human annotations. We plan to do a detailed analysis along two lines: 1) Comparing if the proposed modeling technique can help bridge gap between predicted and human annotations, and 2) Effect of environment variables e.g., background noise, speaker features, different languages etc. We believe our approach can benefit from some straightforward modifications to the architecture, such as using convolutional neural networks which have shown to perform better at handling time-continuous data like speech.

Acknowledgements

This work was supported by the NIH.

References

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*, pages 173–182.

Jie Cao, Michael Tanana, Zac E Imel, Eric Poitras, David C Atkins, and Vivek Srikumar. 2019. Observing dialogue in therapy: Categorizing and forecasting behavioral codes. *arXiv preprint arXiv:1907.00326*.

William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964. IEEE.

Zhuohao Chen, Karan Singla, James Gibson, Dogan Can, Zac E Imel, David C Atkins, Panayiotis Georgiou, and Shrikanth Narayanan. 2019. Improving the prediction of therapist behaviors in addiction counseling by exploiting class confusions. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6605–6609. IEEE.

Yu-An Chung and James Glass. 2018. Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech. *arXiv preprint arXiv:1803.08976*.

Yu-An Chung, Wei-Hung Weng, Schrasing Tong, and James Glass. 2019. Towards unsupervised speech-to-text translation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7170–7174. IEEE.

James Gibson, David Atkins, Torrey Creed, Zac Imel, Panayiotis Georgiou, and Shrikanth Narayanan. 2019. Multi-label multi-task deep learning for behavioral coding. *IEEE Transactions on Affective Computing*.

Albert Haque, Michelle Guo, Prateek Verma, and Li Fei-Fei. 2019. Audio-linguistic embeddings for spoken sentences. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7355–7359. IEEE.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Koji Iwano and Keikichi Hirose. 1999. Prosodic word boundary detection using statistical modeling of moraic fundamental frequency contours and its use for continuous speech recognition. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, volume 1, pages 133–136. IEEE.

J-C Junqua, Brian Mak, and Ben Reaves. 1994. A robust algorithm for word boundary detection in the presence of noise. *IEEE Transactions on speech and audio processing*, 2(3):406–412.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. 2019. Speech model pre-training for end-to-end spoken language understanding. *arXiv preprint arXiv:1904.03670*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

William R Miller, Theresa B Moyers, Denise Ernst, and Paul Amrhein. 2003. Manual for the motivational interviewing skill code (misc). *Unpublished manuscript. Albuquerque: Center on Alcoholism, Substance Abuse and Addictions, University of New Mexico*.

Robert Ochshorn and Max Hawkins. 2016. Gentle: A forced aligner.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE.

Adam Paszke, Sam Gross, Soumith Chintala, and Gregory Chanan. 2017. Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration. *PyTorch: Tensors and dynamic neural networks in Python with strong GPU acceleration*, 6.

Verónica Pérez-Rosas, Rada Mihalcea, Kenneth Resnicow, Satinder Singh, Lawrence Ann, Kathy J Goggin, and Delwyn Catley. 2017. Predicting counselor behaviors in motivational interviewing encounters. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1128–1137.

Yao Qian, Rutuja Ubale, Vikram Ramanaryanan, Patrick Lange, David Suendermann-Oeft, Keelan Evanini, and Eugene Tsuprun. 2017. Exploring asr-free end-to-end modeling to improve spoken language understanding in a cloud-based dialog system. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 569–576. IEEE.

Suman Ravuri and Andreas Stolcke. 2015. Recurrent neural network and lstm models for lexical utterance classification. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Dmitriy Serdyuk, Yongqiang Wang, Christian Fuegen, Anuj Kumar, Baiyang Liu, and Yoshua Bengio. 2018. Towards end-to-end spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5754–5758. IEEE.

Karan Singla, Zhuohao Chen, Nikolaos Flemotomos, James Gibson, Dogan Can, David C Atkins, and Shrikanth Narayanan. 2018. Using prosodic and lexical information for learning utterance-level behaviors in psychotherapy. In *Interspeech*, pages 3413–3417.

Hagen Soltau, Hank Liao, and Hasim Sak. 2016. Neural speech recognizer: Acoustic-to-word lstm model for large vocabulary speech recognition. *arXiv preprint arXiv:1610.09975*.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio, and Christopher J Pal. 2018. Learning general purpose distributed sentence representations via large scale multi-task learning. *arXiv preprint arXiv:1804.00079*.

Michael Tanana, Kevin A Hallgren, Zac E Imel, David C Atkins, and Vivek Srikumar. 2016. A comparison of natural language processing methods for automated coding of motivational interviewing. *Journal of substance abuse treatment*, 65:43–50.

Andreas Tsiartas, Prasanta Kumar Ghosh, Panayiotis Georgiou, and Shrikanth Narayanan. 2009. Robust word boundary detection in spontaneous speech using acoustic and lexical cues. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4785–4788. IEEE.

Gokhan Tur and Renato De Mori. 2011. *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Bo Xiao, Dogan Can, James Gibson, Zac E Imel, David C Atkins, Panayiotis G Georgiou, and Shrikanth S Narayanan. 2016. Behavioral coding of therapist language in addiction counseling using recurrent neural networks. In *Interspeech*, pages 908–912.

Puyang Xu and Ruhi Sarikaya. 2014. Contextual domain classification in spoken language understanding systems using recurrent neural network. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 136–140. IEEE.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Kaisheng Yao, Geoffrey Zweig, Mei-Yuh Hwang, Yangyang Shi, and Dong Yu. 2013. Recurrent neural networks for language understanding. In *Interspeech*, pages 2524–2528.