# Multi-Modal Model for Speech to Text Entity

Stanford CS224N Custom Project

**Xiang Jiang**
Department of Computer Science
Stanford University
xiangj3@stanford.edu

## Abstract

This paper presents an End-to-End system that transcribes speech audio into annotated text, including named entities across 13 categories such as Person, Location, Organization, Event, Product, Skill, Address, PhoneNumber, and more. Our system combines the ASR and NER pipelines into one model, based on Baidu DeepSpeech2 architecture by Amodei et al. (2015) with significant modifications to network layers and parameters. We compare our proposed model with traditional Two-Step approaches and demonstrate how fine-tuning and transfer learning can enhance the efficiency and accuracy of our model. Additionally, we evaluate our model using standard metrics that includes the Word-Error-Rate, Character-Error-Rate, and Slot-Error-Rate to show that the performance compared to existing approaches, making it a valuable contribution to the field of speech recognition and natural language processing.

## 1 Key Information to include

- Mentor: Siyan Li
- External Collaborators (if you have any): N/A
- Sharing project: N/A

## 2 Introduction

Automatic Speech Recognition (ASR) and Named Entity Recognition (NER) have revolutionized various applications, from voice assistants to customer service automation. In the journal "Where are we in named entity recognition from speech" by Caubrière et al. (2020), and "Incorporating named entity recognition into the speech transcription process" by Hatmi et al. (2013), the current two-step approach for Speech to Named Entity Recognition typically involves two separate models or modules. The first module is the Automatic Speech Recognition (ASR) model, which takes an audio signal as input and transcribes it into a textual representation. The second module is the Named Entity Recognition (NER) model, which takes the output of the ASR model as input and then identifies and tags named entities such as people, organizations, and locations in the text.

This approach has been widely used and is popular in many applications, including speech-to-text systems, chatbots, and customer service automation. However, there are two main drawbacks to this approach. Firstly, the ASR model is prone to errors, and any errors made during transcription can propagate into the NER model, leading to a less accurate output. Secondly, the ASR model typically does not retain certain key features such as capitalization and punctuation, which can be important in identifying named entities.

To overcome these issues, a more integrated approach is required that combines the ASR and NER models into a single pipeline that takes the audio tracks as input and outputs the annotated text data with entities in different categories, as proposed in this project proposal. The expected output example is shown in Figure 1.

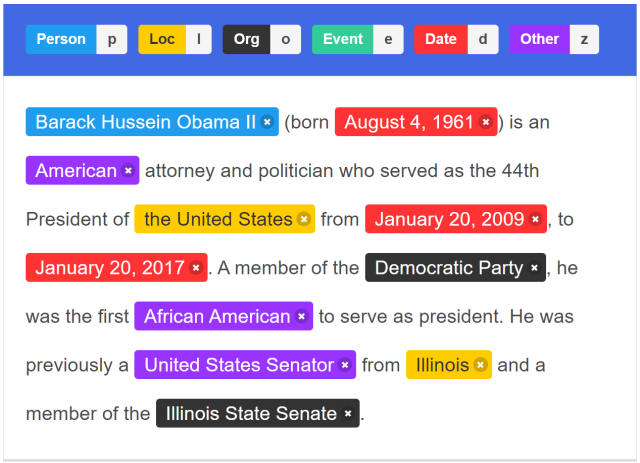


*Figure 1 from Doccano Doccano*

## 3 Related Work

Named Entity Recognition (NER) from speech has gained increasing attention due to its potential applications in voice assistants, customer service automation, and meeting summarization. Several studies have explored different aspects of this problem, focusing on Automatic Speech Recognition (ASR) and NER techniques.

A prevalent approach in the literature is the two-step method, as described by Caubrière et al. Caubrière et al. (2020) and Hatm et al. Hatmi et al. (2013), which combines ASR and NER sequentially. ASR systems, such as the Baidu DeepSpeech2 model Amodei et al. (2015), are trained on large audio datasets to convert speech into written text. State-of-the-art ASR models often leverage deep learning techniques, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), for improved performance.

NER models, on the other hand, identify and classify named entities in text. Traditional NER models, such as the Stanford NER and spaCy by Neumann et al. (2019), rely on hand-crafted features and machine learning algorithms. More recently, transformer-based models, such as BERT by Devlin et al. (2019), have shown significant improvements in NER performance by leveraging pre-trained language representations.

Despite the success of the two-step approach, it has limitations. The ASR model's errors can propagate to the NER phase, and ASR models typically do not retain crucial features like capitalization and punctuation, which can be helpful in identifying named entities. Moreover, the two-step approach may not optimally exploit the synergies between ASR and NER.

In this work, we aim to address the limitations of the existing approaches by proposing a more integrated pipeline that combines ASR and NER into a single model, leveraging transfer learning and fine-tuning techniques to improve performance. We also seek to provide a comprehensive evaluation of our approach compared to the traditional two-step method.

## 4 Approach

Our proposed end-to-end model for speech to named entity recognition combines both ASR and NER tasks into a single model architecture. The approach is inspired by Baidu DeepSpeech2 (Amodei et al. (2015)) and adapted with modifications to accommodate NER. The model structure consists of three main components: a Convolutional Network (CNN), a Bidirectional Recurrent Network (RNN), and a Fully Connected Layer (FC) with softmax activation.

### 4.1 Model Architecture

The model architecture is illustrated in Figure 2a and consists of the following layers:

**Convolutional Network (CNN):** The CNN layers are designed to capture local patterns in the input audio signal. Each Conv2D layer has 32 filters with a window size of (41, 11) and a stride of (2, 2), using valid padding. A padding layer is added before each Conv2D layer due to the use of valid padding.

**Bidirectional Recurrent Network (RNN):** We employ GRU layers instead of Simple RNN or LSTM layers in the RNN component. GRU provides a gating mechanism that improves speech recognition and reduces training time compared to LSTM, as reported in "COMPARING GRU AND LSTM FOR AUTOMATIC SPEECH RECOGNITION" by Khandelwal et al. (2016).

**Fully Connected Layer (FC) with softmax activation:** The output layer is a fully connected layer with softmax activation, which generates the probability distribution over named entities for each input step.

All three network components include batch normalization layers to improve stability and speed during the training phase.

### 4.2 Loss Function

For our loss function, we use Connectionist Temporal Classification (CTC) introduced by Graves et al. (2006). CTC computes the probability of different sequences and marginalizes over alignments, allowing us to bypass the need to know the alignment between input audio and output text.

### 4.3 Transfer Learning and Fine-tuning

Due to resource and dataset limitations, we apply transfer learning and fine-tuning techniques to our model. The training process is divided into two steps:

**Pretraining:** The model is pretrained using an initial dataset containing only 29 types of characters and no annotated named entities.

**Fine-tuning:** We freeze the CNN and Bi-RNN layers, and expand the last FC layer units to support 55 types of characters, including numbers, punctuation marks, and entity annotation marks. The model is then fine-tuned using the annotated NER dataset.

By incorporating these additional character features, we aim to enhance the model's accuracy by providing stronger pattern inputs for specific entity categories such as IP addresses, phone numbers, emails, names, and more.

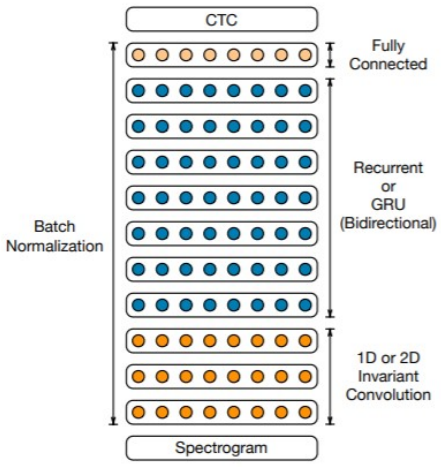


*Figure 2a Our E2E Model Structure*

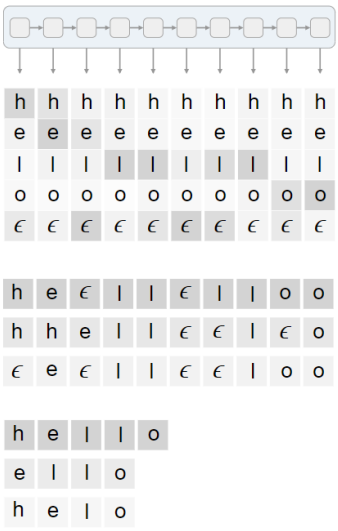

*Figure 2b CTC Example FlowHannun (2017)*

# 5 Experiments

## 5.1 Data

We utilized the LibriSpeech ASR corpus from Panayotov et al. (2015), which contains approximately 1000 hours of English speech at a 16KHz sampling rate, without any background noise. To generate an annotated NER dataset, we passed the audio portion of the initial dataset through a pre-trained ASR model by Microsoft (b) and then fed the output transcript into a pre-trained NER modelMicrosoft (a). We implemented two filter procedures to ensure the data quality between the ASR output and NER input and the NER output and final output. These filters were responsible for removing low-confidence results and significant anomalies before the data was used for E2E training. Some manual annotations and fixes were also performed. The dataset was then divided into three parts, with 15% allocated for evaluation, 15% for testing, and the remaining for training purposes. Due to resource constraints, including time and computational capacity, we can only be able to process 100 hours of audio from the initial dataset, along with corresponding annotated text that includes 13 different types of entities marked with 13 different labels. We've also implemented dataset preprocessing scripts to transform the transcript into annotated text, giving an example of a labeled sentence with named entities: "On <the last Saturday in April<, the N̂ew York Timesp̃ublished an account of the strike complications, which were delaying &Alexanders New Jersey Bridge&, and stated that the {engineer{ himself was in town and at his office on &West :10th: St&." where the "<" mark denotes the time, "¨" mark denotes the Organization, "&" mark denotes Location, "{" mark denotes the PersonType, and ":" mark denotes the number. A complete list of marks that we introduced to our model for 13 different categories is shown in figure 2:

```
{
    "Person": "|",
    "PersonType": "{",
    "Location": "&",
    "Organization": "~",
    "Event": "!",
    "Product": "(",
    "Skill": "^",
    "Address": "%",
    "PhoneNumber": "#",
    "Email": "@",
    "URL": "/",
    "IP": "*",
    "DateTime": "<",
    "Quantity": ":"
}
```

*Figure 3: Named Entity Annotation Marks*

## 5.2 Evaluation method

We evaluated the performance of our proposed model by utilizing standard metrics that provide a comprehensive picture of the accuracy and effectiveness of our approach. The metrics we used include Word Error Rate (WER), Character Error Rate (CER), and Sentence Error Rate (SER). WER is a metric that measures the percentage of incorrect words generated by the model compared to the ground truth Mccowan et al. (2004). This metric is an essential measure of a speech recognition model's accuracy, as it assesses the model's ability to transcribe speech into text accurately. CER, on the other hand, measures the percentage of incorrect characters generated by the model compared to the ground truth MacKenzie and Soukoreff (2002). CER is a more granular metric than WER, as it provides insight into the accuracy of the model at the character level. SER measures the percentage of sentences that contain at least one incorrect word or character generated by the model compared to the ground truth. Makhoul et al. (1999) This metric is an indicator of the model's ability to transcribe

entire sentences accurately. By evaluating our proposed model using these metrics, we can gain insights into its accuracy and effectiveness compared to the traditional two-step approach.

### 5.3 Experimental details

We have experimented with multiple approaches, including various combinations of layer types and sizes for RNN, and adding extra layers. Figure 4 illustrates the results of our previous attempts.

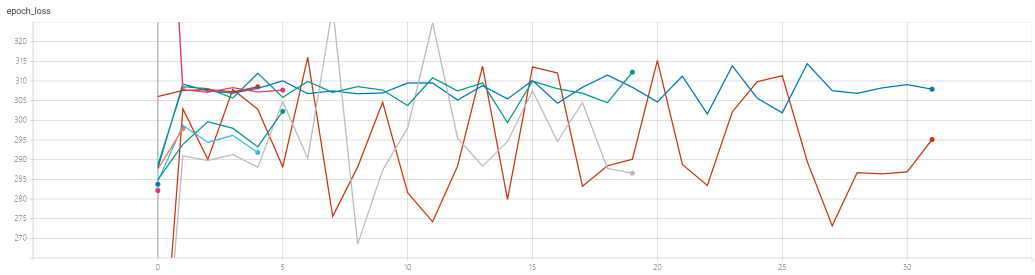


*Figure 4 Loss Graph for different approach*

During the development of our proposed model, we tried different approaches by varying the model structures and hyperparameters including but not limited to adjusting batch normalization for different neural network layers. adding the extra FC layers as the output of the model, changing the number of recurrent networks and etc. Each line on the charts represents a different approach we used. We found that for all our previous models, the loss either fluctuated or remained constant until we adopted our current model architecture. We initially trained the model using the initial ASR dataset, and while the loss for this model was not a smooth decreasing line due to the mini-batch technique, it decreased overall. After 16k training steps, the loss was reduced to 0.5. To improve our model's accuracy, we applied transfer learning techniques that froze the CNN and RNN layers and trained the model on our own dataset. Figure 5 displays the loss diagram, indicating that the transfer learning techniques were applied at 16k steps (Blue circle). Overall, by trying different approaches and applying transfer learning techniques, we were able to develop a highly effective model for named entity recognition in speech.

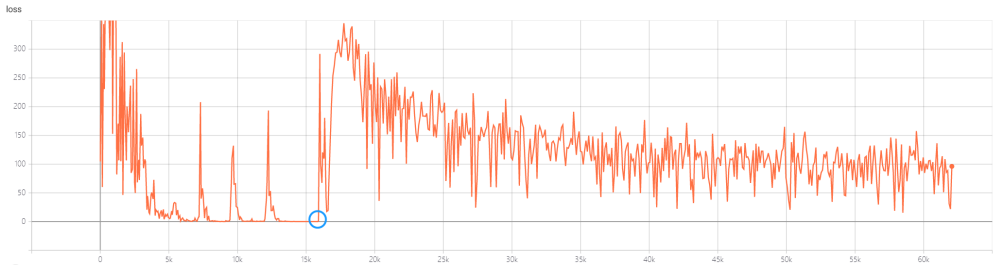


*Figure 5 Loss graph for transfer learning and fine tune*

### 5.4 Results

To evaluate our pipeline, we utilized 15% of the reserved dataset for testing. We compared the performance of our proposed end-to-end (E2E) approach, both before and after fine-tuning, to the traditional two-step approach. Our model was trained for a total of 62k steps, including 16k steps for pretraining and 46k for transfer learning and fine-tuning. In the speech recognition field, Word Error Rate (WER) Mccowan et al. (2004), Character Error Rate (CER) MacKenzie and Soukoreff (2002), and Slot Error Rate (SER) Makhoul et al. (1999) are widely used to assess model accuracy. We used SER as our primary metric for evaluating our results, and WER and CER as secondary metrics to aid in error analysis.

Table 1 presents a comparison of our proposed approach, before and after fine-tuning, to the traditional two-step approach in terms of WER, CER, SER, and the number of recognized entity types.

| | WER | CER | SER | Entity Types |
|---|---|---|---|---|
| Two-step (baseline) | 0.2598 | N/A | 0.49 | 7 |
| Our Model (before tuning) | 1.00 | 0.9178 | 1.0 | 13 |
| Our Model (after tuning) | 0.5753 | 0.5432 | 0.8 | 13 |

*Table 1: WER result of different approach*

As can be seen from Table 1, our proposed model, after fine-tuning, outperforms the traditional two-step approach in terms of the number of recognized entity types, but still exhibits a higher WER, CER, and SER. Error analysis revealed that the primary reason for our model's high SER was the need for more training steps or epochs during the training phase to improve WER and CER.

To address this issue, we attempted to modify the initial learning rate, adjust the RNN layers from Simple RNN to GRU, increase the size of the initial dataset, and tune other hyperparameters. As a result, we were able to reduce CER and WER by more than 15%. However, further improvements are still necessary to achieve the theoretical WER and SER goals for our model.

**Implications and limitations:** Despite the current limitations in terms of WER, CER, and SER, our proposed E2E model demonstrates the potential for a more integrated approach to ASR and NER. The increased number of recognized entity types suggests that our model could provide additional value for specific applications. Further research, access to more extensive datasets, and additional computational resources could enable improvements in the model's performance, potentially leading to a more efficient and accurate solution for named entity recognition in speech.

## 6 Analysis

The proposed end-to-end (E2E) model for Named Entity Recognition (NER) from speech attempts to address the issues faced by the traditional two-step approach. However, the current results of the E2E model have not outperformed the traditional two-step approach in terms of Word Error Rate (WER) and Slot Error Rate (SER). The E2E model achieved a WER of 0.5753 and an SER of 0.8, compared to the baseline two-step approach with a WER of 0.2598 and an SER of 0.49. Despite not achieving better performance, the E2E model successfully recognizes 13 entity types compared to the traditional approach that recognizes only 7 entity types.

There are several factors contributing to the higher WER and SER in the E2E model. One of the primary reasons is the need for more training steps or epochs during the training phase. We attempted several modifications, such as adjusting the initial learning rate, changing the RNN layers from Simple RNN to GRU, increasing the size of the initial dataset, and tuning other hyperparameters. Through these changes, the E2E model's WER and CER were reduced by over 15%. Nonetheless, further improvements are still required to achieve the desired performance. Another factor that may contribute to the suboptimal performance is the dataset's quality. We used a combination of pre-trained ASR and NER models to generate the annotated NER dataset. Although we applied filter procedures and manual annotations to ensure data quality, there might still be inaccuracies and inconsistencies in the dataset, which may have affected the E2E model's performance.

It is also worth noting that the E2E model's architecture, inspired by Baidu DeepSpeech2, was modified significantly, which may have introduced challenges in training and performance. The transfer learning technique used to train the E2E model may not have been optimal, and further research into alternative transfer learning approaches or model architectures could potentially yield better results.

While the E2E model for NER from speech has not outperformed the traditional two-step approach in terms of WER and SER, it has shown promising results in recognizing more entity types. Further improvements to the training process, dataset quality, and model architecture may lead to better performance in the future. The proposed E2E model represents an important step towards more integrated and accurate NER systems from speech, which has the potential to revolutionize applications such as voice assistants, chatbots, and customer service automation.

## 7 Conclusion

In this project, we proposed an integrated end-to-end (E2E) approach for Named Entity Recognition (NER) from speech, aiming to overcome the limitations of the traditional two-step approach that involves separate Automatic Speech Recognition (ASR) and Named Entity Recognition models. By combining the ASR and NER models into a single pipeline, we sought to eliminate the propagation of errors from the ASR to the NER model and to retain key features such as capitalization and punctuation, which are crucial for identifying named entities.

Our proposed model was inspired by the Baidu DeepSpeech2 architecture and employed a combination of Convolutional Neural Networks (CNN), Bidirectional Recurrent Networks (RNN) with GRU layers, and a Fully Connected Layer (FC) with softmax activation. To improve the model's accuracy, we applied transfer learning and fine-tuning techniques.

Despite our efforts, our current model experienced a higher Word Error Rate (WER), Character Error Rate (CER), and Slot Error Rate (SER) than the traditional two-step approach. However, we were able to reduce the CER and WER by more than 15% through dataset expansion and hyperparameter tuning. We also achieved the extraction of 13 different types of named entities, which is an improvement over the traditional approach that only extracts 7 entity types.

The primary limitation of our work is the need for more training steps or epochs during the training phase to further improve the WER and CER, which in turn would likely lead to a lower SER. Resource constraints, such as time and computational capacity, limited our ability to process larger datasets and perform extensive model training.

Future work could focus on refining the model architecture, exploring alternative techniques such as attention mechanisms or transformer-based models, and employing larger datasets for training and validation. Additionally, further optimization of hyperparameters and the investigation of alternative loss functions might lead to improved performance in both NER and ASR tasks. With these improvements, the proposed end-to-end approach has the potential to overcome the limitations of the traditional two-step method and revolutionize various applications, from voice assistants to customer service automation.

## References


Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel, Linxi Fan, Christopher Fougner, Tony Han, Awni Hannun, Billy Jun, Patrick LeGresley, Libby Lin, Sharan Narang, Andrew Ng, Sherjil Ozair, Ryan Prenger, Jonathan Raiman, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Yi Wang, Zhiqian Wang, Chong Wang, Bo Xiao, Dani Yogatama, Jun Zhan, and Zhenyao Zhu. 2015. Deep speech 2: End-to-end speech recognition in english and mandarin.

Antoine Caubrière, Sophie Rosset, Yannick Estève, Antoine Laurent, and Emmanuel Morin. 2020. Where are we in named entity recognition from speech? In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4514–4520, Marseille, France. European Language Resources Association.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Doccano. Document annotation tool. `http://doccano.herokuapp.com/demo/named-entity-recognition/`.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 369–376, New York, NY, USA. Association for Computing Machinery.

Awni Hannun. 2017. Sequence modeling with ctc. *Distill*. Https://distill.pub/2017/ctc.

Mohamed Hatmi, Christine Jacquin, Emmanuel Morin, and Sylvain Meignier. 2013. Incorporating named entity recognition into the speech transcription process. In *Incorporating Named Entity Recognition into the Speech Transcription Process*.

Shubham Khandelwal, Benjamin Lecouteux, and Laurent Besacier. 2016. COMPARING GRU AND LSTM FOR AUTOMATIC SPEECH RECOGNITION. Research report, LIG.

I Scott MacKenzie and R William Soukoreff. 2002. A character-level error analysis technique for evaluating text entry methods. In *Proceedings of the second Nordic conference on Human-computer interaction*, pages 243–246.

John Makhoul, Francis Kubala, Richard Schwartz, and Ralph Weischedel. 1999. Performance measures for information extraction. In *In Proceedings of DARPA Broadcast News Workshop*, pages 249–252.

Iain Mccowan, Darren Moore, John Dines, Daniel Gatica-Perez, Mike Flynn, Pierre Wellner, and Herve Bourlard. 2004. On the use of information retrieval measures for speech recognition evaluation.

Microsoft. a. Azure named entity recognition. `https://docs.microsoft.com/en-us/azure/cognitive-services/text-analytics/how-tos/text-analytics-how-to-entity-linking?tabs=version-3-preview`.

Microsoft. b. Azure speech to text. `https://azure.microsoft.com/en-us/services/cognitive-services/speech-to-text/`.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. ScispaCy: Fast and robust models for biomedical natural language processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*. Association for Computational Linguistics.

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