A NEURAL NETWORK BASED RANKING FRAMEWORK TO IMPROVE ASR WITH NLU RELATED KNOWLEDGE DEPLOYED

Zhengyu Zhou1, Xuchen Song2, Rami Botros3, Lin Zhao2
1 Bosch Research and Technology Center North America, Sunnyvale, CA, USA; 2 ByteDance AI Lab, Palo Alto, CA, USA; 3 Google Inc., Mountain View, CA, USA

Introduction

Previous efforts to improve Automatic Speech Recognition (ASR): various detections explored: end-to-end training, language modeling, etc.

• One popular choice: post-processing hypotheses generated by ASR
• Mainly due to the convenience of applying additional knowledge
• Methods include confusion networks, re-ranking ASR hypotheses with various language/discriminate models, e.g., by classification using SVM or neural network encoder based classifier, etc.

• From the aspect of knowledge usage, limited linguistic knowledge is used

Natural Language Understanding (NLU) information, such as slots and intents, are typically not used in efforts to improve ASR.

Contribution of this work:

• Propose a new framework that uses neural network to simultaneously rank multiple hypotheses generated by ASR for a speech utterance.
• The framework uses all competing hypotheses as input, and predicts the ranking of them (based on scores of corresponding output nodes).
• NLU knowledge is deployed in the framework to facilitate the ranking.

Proposed Framework

Proposed Framework Structure

We propose a new framework, which is a deep feedforward neural network, to simultaneously rank multiple ASR hypotheses for a speech utterance:

• Input: features from N (N>10) in this study) competing hypotheses

• Output: ranking result, optionally together with intent detection result

• The target distribution reserves the ranking information, generating a higher score for an output node if the corresponding input hypothesis contains less ASR errors.
• By minimizing the Kullback-Leibler Divergence loss, the output distribution approximates the target distribution.

The output layer optionally includes an “intent output” part for joint training:
• It could be beneficial to jointly train the ranking task and intent detection, since intent information can offset ASR errors.
• “Intent” related output nodes are corresponding to possible intents, assigned with one-hot target values, and trained with cross-entropy loss.

For the standalone ASR case, training is derived from the ASR ranking part and the intent related part to the lower layers.

Features

Four types of features are extracted in this study:

• Trigger Feature
• Trigger features are used to model long/flexible-distance constraints
• Trigger a pair of linguistic units (i.e., word-slot) that are significantly correlated in a same sentence (e.g., “use BBC”).
• Selection of triggers: Given an in-domain text corpus, use slots to replace corresponding text (e.g., using “song name” to replace “Poker Face”), calculate mutual information (MI) scores of all trigger pairs in the corpus as follows (Rosenfeld, 1994): The top trigger pairs with highest MI scores are then selected as trigger features.

• Extraction of trigger features: Use a standalone NLU module (see next figure) to detect the slots in each hypothesis. The value of a trigger feature is 1 if the trigger pair appears in the hypothesis, and 0 otherwise.
• Trigger feature extraction may help distinguish among the hypotheses.

• BLSTM Feature
• In the extraction of trigger features, the standalone NLU module is used to detect the triggers. It utilises bidirectional LSTM RNN to encode the hypothesis, where the last states of both the forward and backward RNN cover information of entire hypothesis. We concatenate the two last states into a single embedding vector, referred to as the BLSTM features.
• Since the NLU module is a joint model of intent detection and slot filling, the BLSTM features are intent-sensitive.

• Confidence Feature
• Sentence-level confidence score assigned by ASR engine to each hypothesis is also used as feature, and directly fed into the inner layers.

• Note: For a feature type that extracts hundreds or more features per hypothesis, two to simultaneously rank multiple ASR hypotheses for a speech utterance.
• A popular type information to facilitate learning, especially when only ASR output is considered.

• Joint Framework: jointly trained with ASR ranking and intent detection

For either framework, when it predicts the top-ranked hypothesis, it also retrieves that hypothesis’ NLU results (obtained in feature extraction). In this way, ASR and NLU results are obtained simultaneously.

• Adopting the proposed soft target values for ranking is found important. For example, in the joint framework, when replacing the soft target values with one-hot values, the WER obtained rises to 7.21%.

• All the four types of features are beneficial in this case. For example, removing the trigger features (NLU relevant) from the joint framework increases the WER to 7.28%.

• For the standalone NLU module, feeding in named-entity features is beneficial, e.g., reducing the intent error from 9.12% to 5.17% and raising slot filling F1 from 65.45 to 90.66 on testing references.

Experiments

Dataset 1: internal co-in-traffic infotainment corpus (driver assistant tasks)
Training/testing sets: 9166, 1970 utterances respectively.
Speech recorded in car with relatively low noise conditions from multiple speakers with balanced gender.
Decoded by three complementary ASR engines (general ASR engine, plus grammar based and statistical LM based domain-specific engines) Top-best hypothesis from each engine fed into the framework to rank.

Dataset 2: infotainment data.
In this dataset, frame is from 16 name lists, some of which are large (e.g., the song list has 5232 entries). | 40 slot labels (including phone number, frequency, etc. which have no predefined list) were created following IOB schema [Rambani, 1999] | 89 distinct intents (e.g., “tune radio frequencies”) are used.

Dataset 2: public ATIS corpus (airline travel information)
Training/testing sets: 4497, 491 and 893 sentences respectively.
Speech utterances synthesized with noises and reverberation added.
Decoded by one ASR engine: state-of-the-art Google cloud asr, which generates 10 hypotheses at maximum per utterance.

There are 127 slot labels and 18 distinct intents in total.

For both datasets, we develop and evaluate the systems in similar ways.

• NLU module: Trained with references, using GloVe word embedding.
• Decoding by one ASR engine: state-of-the-art Google cloud asr.

Results on ATIS data

Results on In-Car Infotainment Data

• On ATIS, all types of features are beneficial except the BLSTM feature (possibly because competing hypotheses are generated by one engine and are highly similar in sentence embedding). The BLSTM feature is thus removed from the ranking frameworks. Results are shown above.

• ASR-alone framework achieves the best performance, obtaining 21.9% relative decrease in WER over state-of-the-art Google ASR.

• Joint training with intent brings no improvement in this case, also due to the similarity among hypotheses, which typically bear the same intent.

• Adopting the soft target values is also critical. Using one-hot target values instead in the ASR-alone framework leads to a WER of 6.68%.

Summary

We proposed a new framework to rank competing hypotheses for a speech utterance, showing effective no matter the hypotheses are generated by multiple ASR engines or one engine.

• The framework simultaneously generates new ASR and NLU results, improving only ASR but also NLU related features, generating new ASR and NLU results at the same time.

• Novel soft target value is proposed to effectively train the framework.

• Joint training with the framework in the downstream detection is found beneficial when different types of engines are used in decoding.

• For the standalone NLU module, incorporating name type information improves NLU performance when predefined name lists are available.

• In future, further improvements can be made in various directions, such as introducing new features and improving the network design.