Extended Hotword Detection to Arbitrary Audio Triggerwords

https://youtu.be/FdAuHN8v9R8

Kevin Yeun, kyeun@Stanford.edu

Stanford University, CS 230

Summary

Here we explore an extended sequence model approach to detect whether a model trained on a supervised speech recognition task (recognize a subset of words) can be extended to measure similarity between arbitrary audio clips of spoken words (not from the original subset). There exist very accurate and performant hotword detection models today which are modeled as supervised binary classification problems. A generalized model could provide very large utility to smaller independent companies and applications to use their own hotwords or for users to specify hotwords which are more natural in their native language.

The extended model converts 1 second audio spectrograms into a 34 dimensional softmax output corresponding to 34 different English words from the input training set. Then the cosine similarity is measured between the target hotword softmax output and the test hotword softmax output in this new basis (vector space) to determine whether the two audio inputs are equivalent.

Data

The Google Speech Commands dataset [1] is used, consisting of 105,000 16kHz WAVE audio files of 34 different words. Each file is labeled with the true word and is approximately 1 second in length each.

Features

Each 1 second audio clip is preprocessed and converted into a 119 timestamp, 134 frequency spectrogram before fed into the model. Convolutional Neural Networks for Small-footprint Keyword Spotting [2] uses this approach with 2 dimensional convolutions to make the model invariant to time and speaking style. It also reduces the input sequence length to the model from 16000 (16kHz) to 119, improving the model's ability to retain "memory" over the input.

Model

We use a deep learning sequence model consisting of the following layers: Conv1D(196 filters, kernel size=15, stride=4), BatchNormalization, RELU activation, Dropout(0.8), LSTM(128), Dropout(0.5), LSTM(128), Dropout(0.5), Dense, Softmax activation, with a total of 696,918 trainable parameters. The model is built in Keras on Tensorflow [3] and is heavily inspired by the models used in the "Sequence Models" deeplearning.ai Coursera course [4] in "Emojify" and "Triggerword Detection".

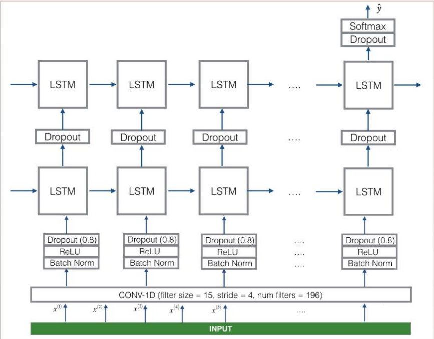

Figure 1: Deep learned sequence model

We then use the softmax output as input to the cosine similarity measurement for classifying the hotword detection. As this function should be available for any arbitrary base hotword, this function is not trained but takes a hyperparameter threshold **α** to determine equivalency.

Results

The sequence model is trained and evaluated using a training set size of 33861 and a test set size of 1292, with Atom optimizatation (learning rate = 0.01, beta1 = 0.9, beta2 = 0.999, decay = 0.01), batch size of 1000, and categorical crossentropy loss. As a baseline, the cosine similarities are measured for the test set on the extended model.

Epochs	Train loss / accuracy (SM)	Test loss / accuracy (SM)	Mean / Variance Test Cosine similarity (same word, EM)	Mean / Variance Test Cosine similarity (other words, EM)
10	0.8811 / 0.7399	2.383 / 0.6076	0.8529 / 0.07874	0.04275 / 0.01168
50	0.5071 / 0.8407	2.667 / 0.6494	0.8975 / 0.06404	0.02779 / 0.008665
100	0.3681 / 0.8815	2.918 / 0.6594	0.9059 / 0.06291	0.02469 / 0.008303

Figure 2: Results after N epochs of Sequence Model (SM) and Extended Model (EM).

The extended model was then evaluated on audio pairs of a new word: e.g. "Stanford", which had highest basis probabilities for words "Sheila" and "one", obtaining overall cosine similarity of **0.6696**.

Discussion

The sequence model itself performs reasonably obtaining test word recognition accuracy 0.6594 of after 100 epochs. Additionally the cosine similarity as expected shows significant correlation between softmax outputs on the test words from the dataset. The cosine similarity of base and eval instance of "Stanford", which was not a word from the training dataset, also looked promising with high cosine similarity of 0.6696. This result supports the hypothesis that similar sounding words which are not from the training dataset could be supported by the extended model. It also suggests non-verbal utterances could be supported as long as the sound can be reconstructed in the input word vector space.

Future

Future work includes exploring more specialized similarity functions for this task or finding more optimal word bases, either by dimensionality reduction (SVD) or involving acoustical characteristics of the words. Also, more datasets should be used to evaluate the extended model.

References

[1] P. Warden, *Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition.* TensorFlow documentation, 2018.
[2] T. Sainath, C. Parada, *Convolutional Neural Networks for Small-footprint Keyword Spotting*. Interspeech, 2015.
[3] Keras.io, 'Keras: The Python Deep Learning Library', 2018. [Online]. Available: https://keras.io/
[4] Coursera.org, 'Sequence Models', 2018. [Online]. Available: https://www.coursera.org/learn/nlp-sequence-models