Development of voice command recognition software for smart-home applications

Abstract: In this study, a word-based speech model was evaluated for the purpose of offline word recognition for thirteen pre-selected spoken commands. An MLP classifier trained on context-aware MFCC and MFCC delta features has achieved an average of 96% accuracy when trained and evaluated on data from sixteen distinct participants. These findings suggest that this approach could be easily adapted to online system in order to obtain an effective solution to the posed problem.

Introduction

The aim of this study is to evaluate the feasibility of a voice command-based system controlling various aspects of modern smart household. One of the advantages of such system is its ease of use (especially by the elderly) and personalization based on user voice sample.

Methods

Sixteen volunteers (aged 22-25; 6 male) were asked to prepare four recordings of themselves reading a script with thirteen pre-selected words that could be used to control a smart house. None of the participants reported having a speech impediment. Participants were asked to label commands in recorded waveform on their own.

Several models were trained and evaluated. Single-subject models were trained on two recordings of a given participant and tested on the remaining two. Multi-subject model was trained using a 70-30 data split.

Following a similar approach to [1], we assumed a simple speech model. However, instead of using a phoneme-based model, each target word was decomposed into three underlying states: <WORD START>, <WORD MIDDLE> and <WORD END>, based on word's length. As an example, the word SWIATLO is decomposed as [(SW), (I), (A), (T), (LO)] which in turn is modeled as the sequence [<SWIATLO START>, 3x<SWIATLO MIDDLE>, <SWIATLO END>]. Additionally, an auxiliary target '-' (PAUSE) was added for the absence of any spoken words. PAUSE was not decomposed into states.

Input waveforms were split into labeled segments. MFCC features [2] of length 13 were calculated in each 25 millisecond recording window, slided with 10 millisecond overlap. Additionally, MFCC delta features [3] were computed over 10 neighboring frames, totaling a feature vector of length 26 for each frame. Finally, target variables were associated with frames and neighboring eight frames, representing a context for each frame. This 26x9 feature matrix was used during model training procedure. Prior to training, features were standardized.

MLP classifier (256 neurons, Adam optimizer, 30% of data used for in-training validation and early stopping) was trained to estimate word-state probabilities. For offline prediction, classifier outputs for each frame in the word waveform are concatenated and the final decision is the mode of predictions, where for the purpose of this voting procedure, word-states are converted back to the initial word.

Findings

Table 1. reports test set classification accuracy for each of the per-subject model.

Table 1. Mean prediction accuracy for models trained on single subject data (separate model for each subject).

Subject ID	Prediction accuracy [%]
258118	84.00
258126	92.31
258135	100.00
266701	94.00
266702	100.00
266708	98.00
266710	100.00
266711	98.00
266712	90.00
266723	100.00
266725	96.00
266753	94.00
266761	94.00
273352	100.00
273356	100.00
282075	100.00
average	96.27 ± 4.65

Model trained simultaneously on all subjects achieved a mean accuracy of 96.61% over all commands. Figure 1. presents a confusion matrix for this model.

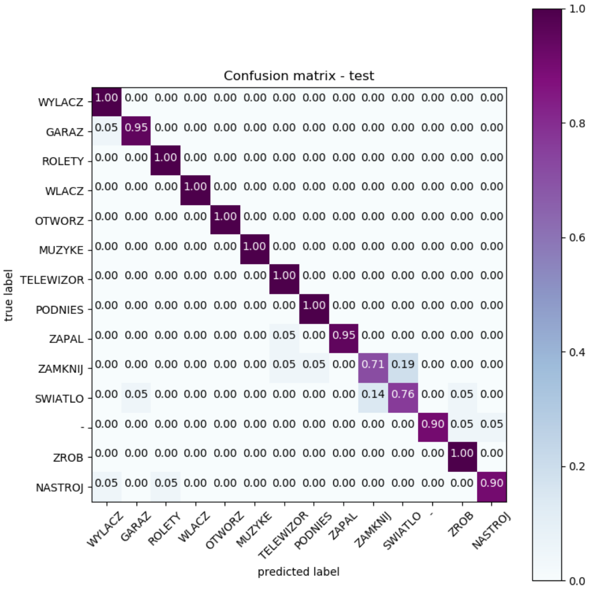

Figure 1. Confusion matrix for multi-subject model.

Discussion

In all tested datasets, a 70-30 train-test split was used. For smaller datasets (four files for single subjects) that ratio was impossible to enforce, and thus 50-50 scheme was employed. In every case split was performed prior to loading any data, which means that test set was independent from training samples (i.e. there exists no overlap between samples in a sliding-window feature extraction paradigm).

Surprisingly, we did not observe higher error and confusion between pairs that are phonetically similar, such as WLACZ-WYLACZ.

It was assessed that feature standardization significantly improved model performance. Having turned standardization off, the resulting classification accuracies were poor at best (around 60% on 'good_quality' dataset, consisting of highly-curated dataset).

Conclusions

Proposed solution, despite not directly taking advantage of temporal correlations present in the signal using more sophisticated models such as HMMs or LSTM, still achieved respectable offline classification accuracy. Moreover, the presented solution seems to be easily adaptable to online prediction.

References

[1] S. Renals, *et.al*, Connectionist probability estimators in HMM speech recognition, IEEE Transactions on Speech and Audio Processing (Volume: 2, Issue: 1, Jan 1994), 161-174

[2] Guide for using MFCC features:

http://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html [Accessed April 2018]

[3] Guide for using MFCC and MFCC delta features:

http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/ [Accessed April 2018]