Semi-Supervised Keyword Spotting in Arabic Speech Using Self-Training Ensembles

Mohamed Mahmoud (elgeish@stanford.edu)

Abstract

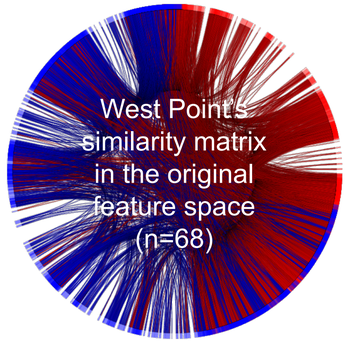
The Arabic speech recognition field suffers from the scarcity of properly labeled data. We introduce a pipeline that performs semi-supervised segmentation of audio then–after hand-labeling a small dataset–feeds labeled segments to a supervised learning framework to select, through many rounds, an ensemble of models to infer labels for a larger dataset; using which we improved the **F1 score of KWS from 75.85%** (using a baseline model) **to 90.91%** on a ground-truth test set. We picked the keyword *na`am* (*yes)* to spot. We define the system's input as an audio file of an utterance and the output is a binary label: keyword or filler.

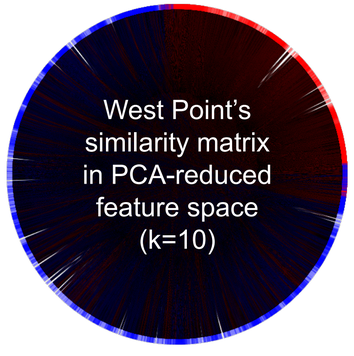
Corpora

Selected utterances from West Point's and King Saud's (unlabeled) corpora were segmented and normalized. We hand-labeled the former and a small subset of the latter; both sets took turns as either a training or a test set for the other but were never mixed. White noise was added to enrich the former set and generalize better.

Features

Each file is divided into short-term frames with sliding windows; we extract 34 features for each frame, from which we derive mid-term windows' features (μ and σ). We use zero crossing rate; energy; entropy of energy; spectral centroid, spread, entropy, flux, and roll-off; 13 MFCCs; and a 12-element Chroma vector and its σ.



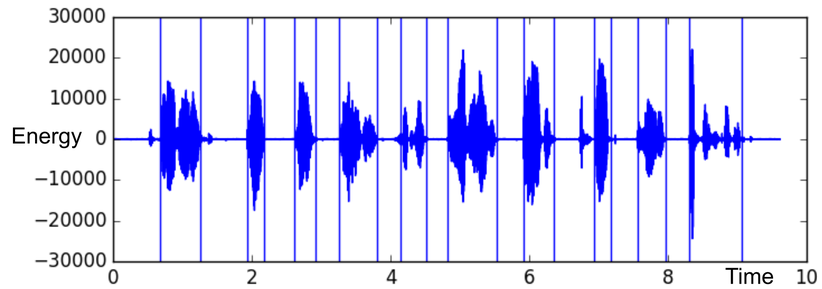


Models Pipeline

West Point's Corpus | King Saud's Corpus

Hand-tuned Semi-supervised SVM

Analyze & Segment | Analyze & Segment



Normalize (mono, bitrate 354K, sampling rate 22050 Hz)

Use Heuristics (Index & Length) & Manually Fix Labels for 2 Subsets

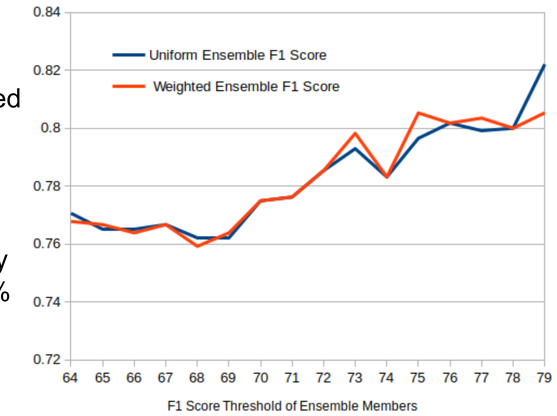
Parameter Sweep Tournament: SVM, RFs, Extra Trees, GB, and KNN

Action	Models	Dataset	Outcome
Train	872	27.8K WP	135 top train-F1 score (bad scores)
Test	135	119 KS	288 (top train- and test- F1 + random)
Train	228	881 WP	50 top train-F1 scores (better scores)
Test	50	119 KS	48 top test-F1 scores (better test scores)
Train	48	1762 (Noisy WP)	8 top train-F1 scores (worse scores)
Test	8	119 KS	Winners are discarded (small test set)
Test	8	587 KS	Winner gets into ensemble rounds
Test	50 (from #3)	587 KS	Winner gets into ensemble rounds
Test	Ensembles Strategies	587 KS	Best F1-score ensemble labels KS
Label	2	62733 KS	Labels are used for training KS
Train	2	15860 Orig. KS	Use results for a baseline
Test	2 + Ensemble	881 WP & 1762 Noisy WP	Best F1-score: 75.85%
Self-Train	2	15860 KS (predicted labels)	Final model
Test	2 + Ensemble	881 WP & 1762 Noisy WP	Best F1-score: 90.91% (Gradient Boosting; single model)
Total	**1220 Unique Models**		

Ensembles

- Uniform Ensemble Prediction = $sign(\sum_i pred_i)$
- Weighted Ensemble Prediction = $sign(\sum_i (pred_i \times F_{1i}))$

Ensemble Analysis: Uniform ensembles outperformed weighted ones as ensemble members improved. They also won when matched against the ground-truth manually labeled datasets: 90% F1 score vs. 79% for the top ensemble of each category.



Final Round's Results

Train (KS Labels)	Test	Predictor	Train F1	Test F1
15680 (predicted)	1762 (WP+Noise)	GB	93.8	**90.91**
15680 (predicted)	881 (WP)	GB	93.8	90.79
15680 (predicted)	881 (WP)	Ensemble	N/A	90.70
15680 (predicted)	881 (WP)	SVM	94.6	89.05
15680 (predicted)	1762 (WP+Noise)	Ensemble	N/A	85.87
15680 (predicted)	1762 (WP+Noise)	SVM	94.6	84.70
15680 (heuristic)	881 (WP)	SVM	82.7	75.85
15680 (heuristic)	881 (WP)	GB	84.1	74.69
15680 (heuristic)	881 (WP)	Ensemble	N/A	71.99
15680 (heuristic)	1762 (WP+Noise)	SVM	82.7	61.82
15680 (heuristic)	1762 (WP+Noise)	GB	84.1	60.98
15680 (heuristic)	1762 (WP+Noise)	Ensemble	N/A	55.17

Discussion & Future Work

- Ensemble learning has a boosting effect for weak learners; self-training helped bootstrap the supervised learning framework; combining them was a big plus.
- KNNs tend to overfit; Extra Trees are slow at scoring.
- Downsampling the negative examples had the biggest impact on improving the results.
- The best performing frame and window parameters are within the literature's recommended range (even for Arabic): 25ms with a step size of 10ms.
- Future Work: Attempts to train neural networks yielded an F1 score of 41% – various setups to test; trying random, instead of grid, search for parameters.

References

[1] *The world factbook*, 2016. [Online]. Available: https://www.cia.gov/library/publications/the-world-factbook/fields/2098.html.

[2] "World economic outlook: Uneven growth—short- and long-term factors.," p. 153, Apr. 2015, ISSN: 1564-5215. [Online]. Available: http://www.imf.org/external/pubs/ft/weo/2015/01/pdf/text.pdf.

[3] J. Benesty, M. M. Sondhi, and Y. Huang, *Springer Handbook of Speech Processing*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007, ISBN: 3540491252.

[4] Y. A. Alotaibi, S. A. Selouani, M. M. Alghamdi, and A. H. Meftah, "Arabic and english speech recognition using cross-language acoustic models," in *Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on*, Jul. 2012, pp. 40–44. DOI: 10.1109/ISSPA.2012.6310585.

[5] *West point arabic speech ldc2002s02*, 2002. [Online]. Available: https://catalog.ldc.upenn.edu/LDC2002S02.

[6] *King saud university arabic speech database*, 2014. [Online]. Available: https://catalog.ldc.upenn.edu/LDC2014S02.

[7] *Timit acoustic-phonetic continuous speech corpus*, 1993. [Online]. Available: https://catalog.ldc.upenn.edu/ldc93s1.

[8] "Audio search based on keyword spotting in arabic language," vol. 5, 2 2014. [Online]. Available: http://thesai.org/Downloads/Volume5No2/Paper_19-Audio_Search_Based_on_Keyword_Spotting_in_Arabic_Language.pdf.

[9] G. Chen, C. Parada, and G. Heigold, "Small-footprint keyword spotting using deep neural networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2014, pp. 4087–4091.

[10] J. R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous hidden markov modeling for speaker-independent word spotting," in *International Conference on Acoustics, Speech, and Signal Processing,*, May 1989, 627–630 vol.1. DOI: 10.1109/ICASSP.1989.266505.

[11] T. N. Sainath and C. Parada, "Convolutional neural networks for small-footprint keyword spotting," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*, ISCA, 2015, pp. 1478–1482. DOI: http://www.isca-speech.org/archive/interspeech_2015/i15_1478.html.

[12] A. Gandhe, L. Qin, F. Metze, A. Rudnicky, I. Lane, and M. Eck, "Using web text to improve keyword spotting in speech," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Dec. 2013, pp. 428–433. DOI: 10.1109/ASRU.2013.6707768.

[13] I. Szöke, P. Schwarz, P. Matějka, L. Burget, M. Karafiát, and J. Černocký, "Phoneme based acoustics keyword spotting in informal continuous speech," in *Text, Speech and Dialogue: 8th International Conference, TSD 2005, Karlovy Vary, Czech Republic, September 12-15, 2005. Proceedings*, V. Matoušek, P. Mautner, and T. Pavelka, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 302–309, ISBN: 978-3-540-31817-0. DOI: 10.1007/11551874_39. [Online]. Available: http://dx.doi.org/10.1007/11551874_39.

[14] B. Kingsbury, "Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, Apr. 2009, pp. 3761–3764. DOI: 10.1109/ICASSP.2009.4960445.

[15] T. N. Sainath, A. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*, 2013, pp. 8614–8618. DOI: 10.1109/ICASSP.2013.6639347. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2013.6639347.

[16] T. Sercu, C. Puhrsch, B. Kingsbury, and Y. LeCun, "Very deep multilingual convolutional neural networks for LVCSR," *CoRR*, vol. abs/1509.08967, 2015. [Online]. Available: http://arxiv.org/abs/1509.08967.

[17] T. Giannakopoulos, "Pyaudioanalysis: An open-source python library for audio signal analysis," *PloS one*, vol. 10, no. 12, 2015.

[18] N. Fazakis, S. Karlos, S. Kotsiantis, and K. Sgarbas, "Self-trained lmt for semisupervised learning," *Computational intelligence and neuroscience*, vol. 2016, p. 10, 2016.