Automatic Classification of Spoken Languages using Diverse Acoustic Features

Yaakov HaCohen-Kerner
Dept. of Computer Science
Jerusalem College of Technology – Lev Academic Center
21 Havaad Haleumi St., P.O.B. 16031
9116001 Jerusalem, Israel
kerner@jct.ac.il

Ruben Hagege
Dept. of Electronics
Jerusalem College of Technology – Lev Academic Center
21 Havaad Haleumi St., P.O.B. 16031
9116001 Jerusalem, Israel
hagege.ruben@gmail.com

Abstract

Many of the language identification (LID) systems are based on language models using machine learning (ML) techniques that take into account the fluctuation of speech over time, such as Hidden Markov Models (HMM). Considering the fluctuation of speech results LID systems use relatively long recording intervals to obtain reasonable accuracy. This research tries to extract enough features from short recording intervals in order to enable successful classification of the tested spoken languages. The classification process is based on frames of 20 milliseconds (ms) where most of the previous LID systems were based on much longer time frames (from 3 seconds to 2 minutes). We defined and implemented 173 low level features divided into three feature sets: cepstrum, relative spectral (RASTA), and spectrum. The examined corpus, containing speech files in seven languages, is a subset of the Oregon Graduate Institute (OGI) telephone speech corpus. Six machine learning (ML) methods have been applied and compared and the best optimized results have been achieved by Random Forest (RF): 89%, 82%, and 80% for 2, 5, and 7 languages, respectively.

1 Introduction

LID is used either as a standalone task or as a pre-processing step, capturing the first seconds (sec) of the recording and processing it in order to transfer the control to the appropriate next stage; e.g. speech recognition systems, multilingual translation systems or call-centers (e.g., emergency calls) routing, where the response time of a native operator might be critical.

LID is a process by which a given spoken utterance language is automatically identified (Muthusamy et al., 1994). Most LID systems are based on high level features such as frequency of a single phoneme, phoneme sequences (Zissman and Singer, 1994), syllable, words, and prosody (Thymé-Gobbel and Hutchins, 1996). Such LID systems need a comprehensive corpus, including transcription from trained humans, and long enough intervals to correctly classify, first, these high level features and then the spoken language (Zissman, 1996; Greenberg, 1999). Any error in the higher level feature recognizers is carried over, and probably/possibly amplified in, the following steps. However, providing a comprehensive corpus enables higher level features which ensure better results than using acoustic features alone. LID systems based on higher level features have one principal problem: Tokenizing those features accurately has proven to be the main obstacle thus far in high accuracy of natural LID (Abramson, 2003). Matejka et al. (2005) found that separating gender before processing improved the LID's accuracy.

A LID system has two main parts: feature extraction, where a vector of measurements that

29th Pacific Asia Conference on Language, Information and Computation: Posters, pages 275 - 285
Shanghai, China, October 30 - November 1, 2015

should characterize the high level features are extracted from the signal; and pattern matching, where these extracted features are processed using statistical (like in this study) or temporal (Rabiner, 1989) methods to recognize speech languages. The approach taken in our study does not resort to the use of phoneme recognizers or any higher level features. Instead, we rely on low-level features alone, rather than using low-level features to predict intermediate features as in previous work. The motivation is "quicker response time and simpler training stages".

The rest of this paper is organized as follows: Section 2 presents an overview of previous LID systems. Section 3 describes the different feature sets chosen for this study. Section 4 presents the suggested classification model and the implemented features for LID of seven languages: French (FR), Farsi (FA), Japanese (JA), Korean (KO), Mandarin (MA), Tamil (TA), and Vietnamese (VI). Section 5 describes the examined corpora and experimental results and analyzes them. Section 6 includes a summary and proposes suggestions for future research.

2 Previous LID Systems

In this section, we focus our overview of previous LID systems that had goals similar to our work or systems that used the same (or a very similar) corpus and / or set of languages.

Silences are an integral part of speech recordings in all languages. These silences are usually unnecessary for computer processing purposes: they considerably increase the files size and potentially lead to a great loss of accuracy of the LID system. Thus, the first step in most LID systems use a Voice Activation Detection (VAD), a sub-process that identifies and discards those silences. Other factors must also be taken in account, such as the channels through which the speech is conveyed. These channels add noises to the speech which, although it is still recognizable by Humans, causes difficulties for computers. Therefore, to ensure better performance using ML methods, a noise-filtering sub-process is preferable. All the previous LID systems described below used at least one of those techniques to enhance their results. Thus, we decided to implement those techniques as well.

Hazen and Zue (1993) tested their system on the OGI Multi-Language telephone speech (MLTS) corpus (Yeshwant K. Muthusamy et al., 1992). Using both genders on the speech utterances. The average length of selected utterance on the OGI corpus is about 13.4 sec. They developed and tested a LID system based on a segment-based approach composed of phonotactic (Matejka et al., 2005), prosodic, and acoustic property of the languages. The features used are 14 Mel Frequency Cepstral Coefficients (MFCC), in contrast to most LID systems that use 13 MFCCs, for each frame. The Cepstral Coefficient (CC) deltas were also extracted along with the pitch (F0) feature, which was used to find and discard silences (VAD) as well as removing the speaker dependency. Each frame was 5ms long. They tested their system on 10 languages, an overall system performance of 48.6% was achieved using n-grams, acoustic, duration, F0, and delta-F0 features. The correct language was one of the top three choices 74.4% of the time. Their results on less than a sec for each file is between 10% and 20%.

Muthusamy et al. (1993) based their system on the OGI-MLTS corpus with 13.4 sec of speech per file on average. They explained that at the time it was still not clear which of the possible LID techniques will be more suitable to discriminate languages. Thus, they compared 3 different approaches (acoustic features, category segmentation, and phonetic classification). In all the sets, the Perceptual Linear Predictive (PLP; Dave, 2013) coefficients was applied using 10ms frames with either 4ms or 7ms of overlapping intervals. Their best result was obtained using 200 bigrams and unigrams. They classified the whole speech files (up to 50 sec) using these feature sets and the Artificial Neural Network (ANN; Lopez-Moreno et al., 2014) ML method. Best results of 86.3% on 2 languages (EN and JA) were obtained. They also obtained 70% accuracy using acoustic features (PLP) alone.

Lamel and Gauvain (1994) presented a LID system tested on the OGI corpus and Laboratory quality speech (four different corpora, two for EN and two for FR language). They applied phone-based acoustic likelihoods, using parallel-trained Hidden Markov Models (HMMs). In 10 languages classification tasks, they tested the OGI corpus and got 48.7%, 55.1%, and 59.7% on intervals of 2, 6 and 10 sec, respectively. On 2 languages (FR and EN) however, their results rose to 76%, 80.87%, and 81.33% on 2, 6, and 10 sec, respectively.

Shuichi and Liang (1995) tested their system on corpora produced from multiple respected sources,

containing the OGI, NTT and NATC corpora. They proposed a LID system based solely on F0 and its time-dependent patterns using discriminant analysis on the polygonal line approximation of the F0 patterns. Using the 21 extracted features from the F0 behavior (e.g., slope, shape, etc.) They achieved 75% on the NTT and NATC corpus and 63.3% on the OGI corpus.

Zissman (1996) compared different LID techniques on the OGI corpus. he also uses RelAtive SpecTrAl (RASTA; Hermansky and Morgan, 1994) as a part of the pre-processing of speech in order to remove slowly varying, linear channel effects from the raw feature vectors. He obtained that single-language phone recognition followed by language-dependent language modeling (PRLM) gave best results when distinguishing 10 languages, giving results as high as 79% on 45 sec speech utterances and 63% on 10 sec. Furthermore, their results in 2 languages discrimination were up 97% on 45 sec of speech (EN and SP) using parallel phone recognition (PPR; Nagarajan and Murthy, 2004) and 90% on 10 sec (JA and SP) using parallel PRLM, they also tested Gaussian Mixture Model (GMM) achieving 84% on 10 sec long audio file (EN and JA).

Lippmann (1997) compared human and state of the art LID available at the time and noted that even if machine ability to identify a language was still several order of magnitude lower than human, he only proved that it was needed to work on more reliant, noise robust, LID systems and components. "The transcription error rate (ER) is less than 0.009% for read digits, less than 0.4% for read sentences from the Wall Street Journal, and less than 4% for spontaneous conversations recorded over the telephone." His study was focused more on isolated digits or alphabet letters recognition in order to perform LID than spontaneous conversation.

Pellegrino and Andre-Obrecht (2000) tested a LID system on 5 languages from the OGI-MLTS corpus: FR, KO, VI, JA, and SP. Using two different approach (GMM and HMM) to model either the vocalic (GMM) or phonetic (HMM) space. Features such as MFCC (8 coefficients) and duration of the segments obtained using a so called "Forward Backward Divergence" (Andre-Obrecht, 1988) segmentation algorithm. The features are extracted inside segments by frames of 20ms. The purpose of this study was to demonstrate the possibility to extract vowel information from acoustic signal. Results were presented either in segments of 2 minutes or 45 sec of speech. Their best results are 73.8% and 61.2% on 4 and 5 languages, respectively, using 2-minute-long speech utterances and all of the features presented earlier.

Kirchhoff and Parandekar (2001) based her LID system on the OGI corpus. Using Multi-Stream Statistical N-Gram Modeling, he compared the accuracy of the model on different speech lengths (from 3 to 45 sec). Features such as manner, consonantal place, vowel place, front-back, and rounding and their dependencies (front-back -vowel place and front-back – consonantal place) were used. On 10 languages, her results were as high as 48%, 58.8%, and 64.6% on audio files of less than 3 sec, between 3 and 15 sec, and longer than 15 sec audio files respectively.

Torres-carrasquillo et al. (2002) used the 1996 Linguistic Data Consortium's CallFriend LID evaluation set, a 12 languages corpus that was allocated as follows: The development set consists of 1184 30-sec utterances and the evaluation set of the corpus consists of 1492 30-sec utterances, each distributed among the various languages of interest. LID was performed using GMM Tokenization: extracting features to then tokenize them using GMM and finally perform LM (in an attempt to enhance the PRLM system developed by Zissman in 1996). Using the evaluation set, an ER of 17% (83% of accuracy) was obtained using both Parallel-PRLM, GMM tokenizers, and GMM acoustics.

Li et al. (2007) investigate automatic spoken language identification (LID) process based on Vector Space Modeling (VSM; e.g., Martínez et al., 2011). The evaluation is carried out on recorded telephone speech of 12 languages: Arabic, EN, FA, FR, GE, Hindi (HI), JA, KO, MA, SP, TA, and VI from 1996 and 2003 NIST Language Recognition Evaluation. Achieving ER as low as 2.75% (97.25% of accuracy) on 30-sec of speech on 6 languages identification. The 2nd focus in their project was the possibility of Real-time (RT) applications.

All those studies based their performance evaluation on a wider time frame than ours, this is a major difference, and it must be considered when comparing our results. Moreover, unlike most of the previous works, our system is not designed to classify languages using keyword, phoneme, or even vowel recognition. It doesn't require any language model either, making the language training process a lot faster.

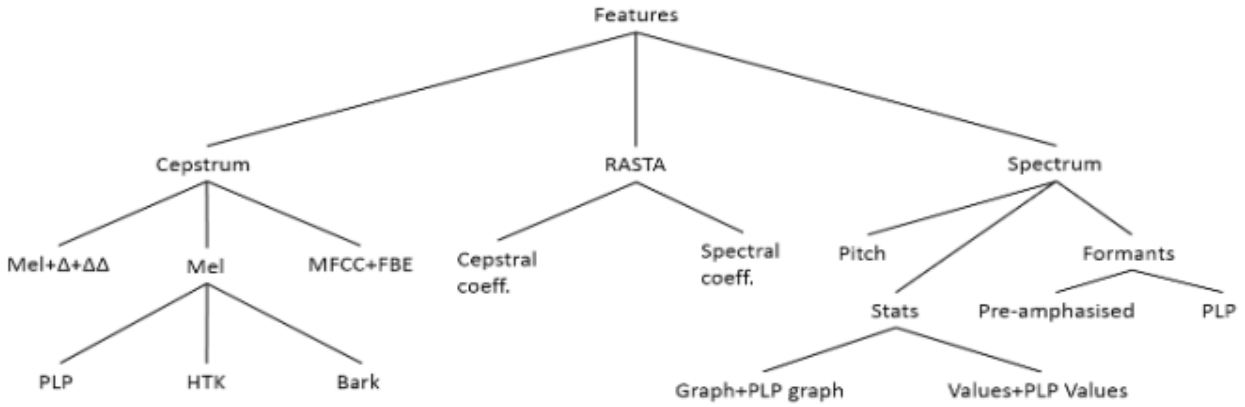

Figure 1. The computed acoustic features.

3 Acoustic Features

In this research, we consider 173 acoustic features divided into three main feature sets: 114 Cepstrum features, 28 RASTA features, and 30 Spectrum features. The hierarchical structure of the three feature sets is described in Figure 1. Although most of these features have been extensively used in previous LID systems, these features were a basis for higher level features. In contrast, our system is solely relying on an extensive combination of low level features which has never been used before to the best of our knowledge.

The Cepstrum features set is composed of groups of coefficients which represent the filter sources (e.g., shape of the mouth etc.). The Bark and Mel scales (Stevens et al., 1937; Stevens and Volkmann, 1940) are perceptual scales of the pitch. Filter Bank Energy (FBE) represents the energy from all the band filters (Huang et al., 2001) used to extract the MFCCs. HTK (HMM ToolKit) represents the CCs extracted using parameters close to the original HTK (Young et al., 2002; Ellis, 2005; Brookes, 1997) approach.

The RASTA set represents features extracted after filtering. These features are extracted in both spectrum and cepstrum, taking cepstrum coefficients using both Linear Predictive Coefficients (LPC), which are used to compute spectral and cepstral features, and RASTA filter.

We implemented the IIR RASTA filter as it is described in Equation 1 (Ellis 2005; Matlab RASTA's filter transfer function implementation).

$$H(z) = 0.1 \times \frac{2z^5 + z^4 - z^2 - 2z}{z - 0.94} \quad (1)$$

The -0.94 weight in the denominator side was chosen in our Matlab implementation to improve filter response time from the original 500ms to 160ms response time using -0.98 that is applied in some of the previous works (Zissman, 1996).

The Spectrum features set consists of the following feature sets: (1) The pitch (F0) feature (Titze, 1994; Zahorian and Hu, 2008). (2) The graph features, which are statistical features that record the occurrence of each frame's median peak. (3) Values (mean, median, min, max, std), and frequency (median) stats, describing each frame's FFT. (4) Formants are the principal spectral component of a frame, defined by "the spectral peaks of the voice spectrum". Linguists largely maintain that the first two formants (in EN at least) are sufficient to differentiate between all vowels (Ladefoged and Johnson, 2014). We decided to extract the 4 first formants.

There is a spectral tilt in speech caused by the voice-source (vocal tract). The vocal tract creates the formant frequencies, so when these are estimated (using FFT), the spectral tilt needs to be removed. This is usually done with a simple pre-emphasis filter, as in our case.

The algorithms that were developed, using MATLAB (V8.3), for this study were built for feature extraction, VAD, and WEKA interfacing purposes. They were designed to perform for real-time applications and, in addition, to be dynamic so that they could be easily changed to extract any specific set of features and/or classes. WEKA (Hall et al., 2009) explorer was used for the classification task.

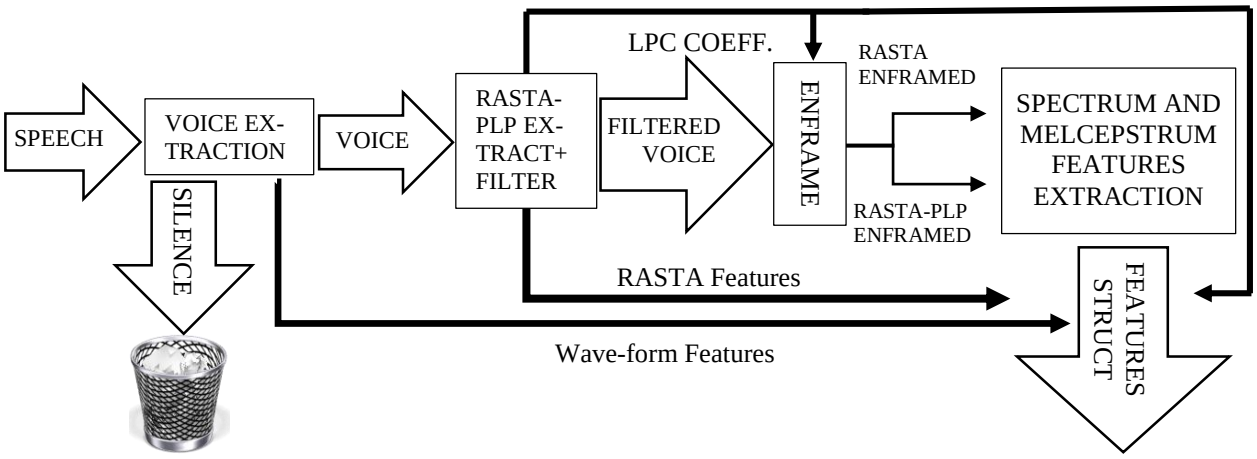

Figure 2. The feature extraction process (stages 2-3 in the classification model).

4 The Classification Model

The main stages of the classification model are as follows:

1. Building the speech corpus (Table 1).
2. Cleaning the speech files. Removing the silent intervals and filtering each file (Figure 2).
3. Computing the features for each file (Figure 2).
4. Transforming the features matrix into a WEKA input file.
5. Applying six ML methods on various combinations of feature sets using WEKA.

Figure 2 describes the feature extraction process (stages 2-3 in the classification model). This Figure grossly illustrates how the structure containing the features, used to discriminate the languages, is extracted. In order to process the speech files as clean as possible equalization and filtering seemed appropriate to better distinguish noise or silence from speech (experimentation shows an improvement of at least 5% in VAD classification after RASTA filtering compared to before).

A RASTA filter is applied to suppress the effect of the telephone line on the features. First, the audio file (speech) is passed through a VAD, and the silence intervals are discarded. One of the features used to perform the VAD (F0) is also extracted (Zahorian and Hu, 2008a). Speech, rid of silences, goes through RASTA feature extraction that extracts the RASTA features family and filters the audio files. The filtered, silence-free speech file is then enframed (Brookes and others, 1997) into frames of 20ms with 10ms overlap, and a Hamming window is applied on each frame (where the last frame is discarded if shorter than 20ms). The frames are sent to the spectrum and cepstrum features extraction where remaining features are extracted. Then, the features extracted are grouped together inside a "features structure" with each frame's features contained in a single line vector. Every file, after completing the feature extraction process, outputs a structure composed of X vectors (depending on file length) containing the 173 features. The resulting structure is then converted into a matrix, and the matrices are concatenated so that every language gets a part of all the files (presented experimented on gets 10,000 feature vectors (frames) for each language.

Six supervised ML methods including one decision tree, two ensemble learning, and two SVMs, have been selected for application of the last stage in our model:

1. J48 is an improved variant of the C4.5 decision tree induction (Quinlan, 1993; Quinlan, 2014) implemented in WEKA. J48 is a classifier that generates pruned or unpruned C4.5 decision trees. The algorithm uses greedy techniques and is a variant of ID3, which determines at each step the most predictive attribute, and splits a node based on this attribute. J48 attempts to account for noise and missing data. It also deals with numeric attributes by determining where thresholds for decision splits should be placed. The main parameters that can be set for this algorithm are the confidence threshold, the minimum number of instances per leaf and the number of folds for REP. As described earlier, trees are one of the easiest thing that could be understood because of their nature.
2. RF, an ensemble learning method for classification and regression (Breiman, 2001). This ML technique is an ensemble learning

technique. Ensemble methods use multiple learning algorithms to obtain better predictive performance than what could be obtained from any of the constituent learning algorithms. RF is based on what's called a random tree: a tree that randomly chooses K attributes and then build a simple tree with no pruning. RF let us choose the number of features (K) and the number of random trees (I) we want to use.

3. MultiBoostab (MB) (Webb, 2000) is an extension to the highly successful AdaBoost (Freund and Schapire, 1996) technique for forming decision committees. MB technique can be viewed as combining AdaBoost with wagging (Bauer and Kohavi, 1999). It is able to harness both AdaBoost's high bias and variance reduction with wagging's superior variance reduction. Using C4.5 as the base learning algorithm, Multiboosting is demonstrated to produce decision committees with lower error than either AdaBoost or wagging significantly more often than the reverse. It offers the further advantage over AdaBoost of suiting parallel execution. In WEKA, the default base classifier for MB is Decision Stump (Iba and Langley, 1992).
4. BayesNet (BN) is a variant of a probabilistic statistical classification model that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG) (Friedman et al., 2000; Heckerman, 1997; Pourret, 2008).
5. Logistic regression (LR) (Cessie et al., 1992) is a variant of a probabilistic statistical classification model that is used for predicting the outcome of a categorical dependent variable (i.e., a class label) based on one feature or more (Landwehr et al., 2005; Sumner et al., 2005).
6. Sequential Minimal Optimization (SMO; Platt, 1998; Keerthi et al., 2001) is a variant of the Support Vectors Machines (SVM) ML method (Cortes and Vapnik, 1995). The SMO technique is an iterative algorithm created to solve the optimization problem often seen in SVM techniques. SMO divides this problem into a series of smallest possible sub-problems, which are then resolved analytically.

These ML methods have been applied using the WEKA platform (Frank, 2006; Hall et al., 2009). We performed parameter tuning with Info-Gain (IG), a feature selection metric for classification purposes. Yang and Pedersen (1997) reported that IG performed best in their multi-class benchmarks. The accuracy of each model was estimated by a 10-fold cross-validation test.

5 Experimental Results

The OGI Multi-language Telephone Speech Corpus (Muthusamy et al., 1992; Muthusamy et al., 1993) consists of telephone speech recorded in eleven languages: EN, FA, FR, GE, HI, JA, KO, MA, SP, TA and VI. The OGI corpus is not balanced between males and females: the male files represent more than 75% of the corpus. Thus, in this study, we only used the male speech files. The examined corpus contains 478 files (each from a different person) from seven selected languages with an average length of 44.3 sec, each file consists of free, continuous speech.

Since our classification system was heavily consuming a classic workstation's RAM, the final corpus had to be reduced to 10,000 frames per language (equally distributed on the various files), that are equivalent to 100 sec of speech. As most of telephone speech corpus based LID systems (Hermansky, 2011), we used a RASTA filter (Matlab implementation; Ellis, 2005) to reduce the channel (telephone) effect noises.

Table 1 presents general information about the speech files contained in the examined corpus. The number of speech files for each language is ranging from 53 to 86. The average time length is rather similar for all languages (from 42.2 to 47.5 sec).

#	Language	# of speech files	Length of speech files in sec.	Avg. time length in sec.
1	French (FR)	55	37<x<49	47.5
2	Farsi (FA)	81	5<x<49	44.4
3	Japanese (JA)	53	23<x<49	46.6
4	Korean (KO)	62	4<x<49	42.2
5	Mandarin (MA)	73	10<x<49	42.5
6	Tamil (TA)	86	8<x<49	44.3
7	Vietnamese (VI)	68	7<x<49	43.9

Table 1. General information about the speech files selected from the OGI corpus.

#	Languages	BN	SMO	LR	MB	J48	RF
2	FR, TA	66.47	72.59	73.02	66.84	80.21	**88.27**
3	FR, MA, TA	54.25	58.76	60.41	42.96	68.47	**81.17**
4	FR, MA, TA, VI	45.99	50.00	51.04	34.11	62.72	**77.51**
5	FR, FA, MA, TA, VI	36.84	42.81	43.34	27.45	57.03	**73.97**
6	FR, FA, JA, MA, TA, VI	32.36	37.54	37.70	22.89	53.29	**71.83**
7	FR, FA, JA, KO, MA, TA, VI	29.38	33.52	33.66	19.48	51.50	**71.13**

Table 2. Accuracy results for the best language combinations using default parameters and all features.

For each tested combination of feature sets we applied all of the 6 chosen ML methods: BN, SMO, LR, MB, J48 and RF. We then checked our feature sets using IG, among other feature selection methods, and no features with zero weights were found. We also performed a parameter tuning process in order to achieve the best results on the best default ML method (see Figure 3). All the optimized results are obtained as follows: each ML parameter is tuned in a hill climbing fashion, changing one parameter at a time (manually) until the best value is obtained (within a <1% margin). On ML methods based on simple trees such as J48, it appears to be enough: the parameters seemed to be independent (according to the results we had). However, for the RF ML method, the two principal parameters were tuned together since our preliminary results tends to show that they have an influence on one another.

Unlike previously developed methods (see Section 2) that focus on changes of specific features over time to classify languages, our research assess the potential of features computed on a single frame (20ms), using each frames as a basis of the classification decision.

Table 2 presents the accuracy results for the 6 selected ML methods under default parameters proposed by the WEKA platform. The best language combinations from 7 to 2 languages (with accuracy as the deciding factor) were selected by analyzing the confusion matrices that were produced by the best ML method – RF (according to Table 2), and filtering out the less successful language in each stage. Firstly, The RF ML method has been applied on the all seven languages and then the six best languages (achieving the best accuracy) were picked from those seven based on the confusion matrix, and so on, until only the best combination of two languages remains. As a result, we got the following language combinations:

7. FR, FA, JA, KO, MA, TA, and VI.
6. FR, FA, JA, MA, TA, and VI.
5. FR, FA, JA, TA, and VI.
4. FR, JA, TA, and VI.
3. FR, JA, and TA.
2. FR, and TA.

Various conclusions concerning our LID system can be drawn from Table 2: (1) The RF method obtained the best accuracy results. (2) The 2nd best ML method was J48. (3) The decision tree ML methods are the best ML methods for our LID tasks.

Since RF is uncontestably the most suited technique between the six chosen ML techniques, we decided to optimize the RF's parameters (maxDepth, numFeatures, numTrees, and seed). Because of the lack of space to display results, we were only able to present optimized results on a limited set of languages. We chose to optimize the best language combinations of size 2, 5, and 7 (see Table 2). All the optimized results are obtained as follows: each parameter is tuned in a hill climbing fashion. By manually changing one parameter at a time till the best value is obtained within a reasonable (<0.1%) margin.

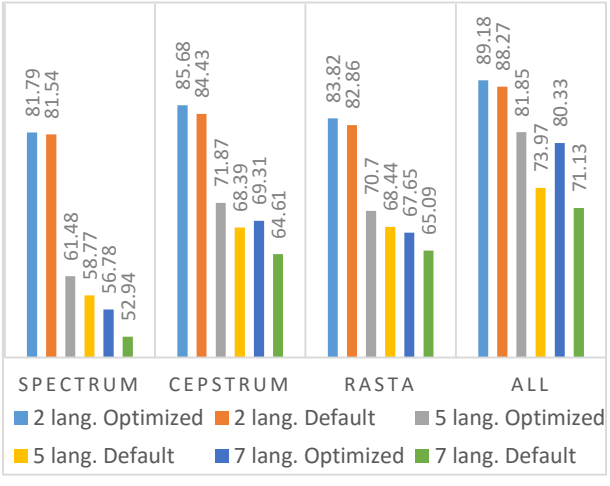

Figure 3. Optimized/default accuracy on each feature set and all features.

Multiple conclusions can be drawn from Figure 3: (1) RF has a great optimizing potential, (2) The more language it classifies, the greater become the optimization over default results, (3) The Cepstrum feature set has the greatest differentiation potential. A possible explanation for these results can be the high number of relevant features: the more relevant data one have, the easier classification become. (4) RASTA has the greatest differentiation potential per feature; its performance is almost equal to the Cepstrum set while using only a quarter of its number of features.

6 Summary and Future Research

In this paper, we present a methodology for classifying speech files from 7 different languages based on combined cepstrum, RASTA, and spectrum feature sets. This methodology compares six different ML methods. RF, the best ML method achieves relatively high accuracy results of 89.18%, 81.85%, and 80.33% for the following classification experiments: 2, 5, and 7 best language combinations, respectively.

The novelties of this research are in its reliance: (1) on low-level features alone, rather than using low-level features changes over time to predict intermediate features as in previous work, and (2) on much smaller frames (20ms) in comparison to most previous LIDs whose results are based on much longer time periods (at least 3 sec. or longer; see Martinez et al., 2013, among many other references below, for detail on the impact of frame length on result). Eliminating reliance on intermediate features is an important contribution, especially for low-resource languages.

Our results are comparable to the accuracy level of top LID systems from about 20 years ago (that also used different versions of the OGI corpus; see section 2). However, our LID system uses a time frame that is at least 60 times shorter than the time frames used by previous LID systems. To the best of our knowledge, there is no LID system which is based on a such short time frame.

Future directions for research are: (1) Developing additional feature sets in general and additional features in particular (with an emphasis on the RASTA set), (2) Applying other ML methods in order to find the most suited method for LID purposes, (3) Conducting more experiments using more speech files from more languages, (4) Discovering which combination of features in particular are appropriate for LID of speech files using the system we developed, and (5) How well does the system based on acoustic features work for non-native speakers?

Acknowledgments

The authors would like to thank Shmuel Kirshner for his many advises on theory of speech processing, Edmond Shalom for enabling us to start this research, Shlomo Engelberg for his continuous support, on each aspect of this endeavor, Shimon Mizrahi for giving us the time needed to accomplish such a work, Boris Dekhovitch for his comments, Evgeni Frishman and Yaakov Friedman for financing of the database. Many thanks to the Dept. of Electronics and the rector Kenneth Hochberg of the Jerusalem College of Technology, Lev Academic Center, for their assistance during this research. We would also like to thank the three reviewers for their useful and instructive comments.

References

Arthur S. Abramson. 2003. *A Practical Introduction to Phonetics (review)*.volume 79. Clarendon Press Oxford.

Régine Andre-Obrecht. 1988. A New Statistical Approach For The Automatic Segmentation Of Continuous Speech Signals. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(1):29–40, January.

Eric Bauer and Ron Kohavi. 1999. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning*, 36(1):105–139.

Leo Breiman. 2001. Random Forests. *Machine Learning*, 45(1):5–32.

Mike Brookes. 1997. Voicebox: Speech Processing Toolbox for Matlab. *... From Www. Ee. Ic. Ac. Uk/Hp/Staff/Dmb/Voicebox/Voicebox ...*.

Saskia Le Cessie, J. C. Van Houwelingen, and Royal Statistical Society. 1992. Ridge Estimators in Logistic Regression. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 41(1):191–201.

Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Machine learning*, 20(3):273–297.

Namrata Dave. 2013. Feature Extraction Methods LPC, PLP and MFCC In Speech Recognition. *International Journal for Advance Research in Engineering and Technology*, 1(Vi):1–5.

Daniel P.W. Ellis. 2005. PLP and RASTA (and MFCC, and Inversion) in Matlab. *Http://Www.Ee.Columbia.Edu/~Dpwe/Resources/Matlab/Rastamat/*.

Joachim Frank. 2006. *Electron Tomography: Methods for Three-Dimensional Visualization of Structures in the Cell*. Morgan Kaufmann.

Yoav Freund and Re Robert E Schapire. 1996. Experiments with a New Boosting Algorithm. In *International Conference on Machine Learning*, volume 96, pages 148–156.

Nir Friedman, M Linial, I Nachman, and D Pe'er. 2000. Using Bayesian Networks to Analyze Expression Data. *Journal of computational biology : a journal of computational molecular cell biology*, 7(3-4):601–620.

Steven Greenberg. 1999. Speaking in Shorthand - a Syllable-Centric Perspective for Understanding Pronunciation Variation. *Speech Communication*, 29(2):159–176.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software. *ACM SIGKDD Explorations Newsletter*, 11(1):10.

Timothy J Hazen and Victor Zue. 1993. Automatic Language Identification Using a Segment-Based Approach. In *3rd International Conference on Spoken Language Processing*, pages 1307–1310.

David Heckerman. 1997. Bayesian Networks for Data Mining. *Data Mining and Knowledge Discovery*, 119(1):79–119.

Hynek Hermansky. 2011. Speech Recognition from Spectral Dynamics. *Sadhana - Academy Proceedings in Engineering Sciences*, 36(5):729–744.

Hynek Hermansky and Nelson Morgan. 1994. RASTA Processing of Speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4):578–589.

and Raj Foreword By-Reddy. Huang, Xuedong, Alex Acero, Hsiao-Wuen Hon. 2001. *Spoken Language Processing : a Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR.

Wayne Iba and Pat Langley. 1992. Induction of One-Level Decision Trees. In *ML92: Proceedings of the Ninth International Conference on Machine Learning, Aberdeen, Scotland, 1–3 July 1992*, pages 233–240.

Sathiya S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy. 2001. Improvements to Platt's SMO Algorithm for SVM Classifier Design. *Neural Computation*, 13(3):637–649.

Katrin Kirchhoff and Sonia Parandekar. 2001. Multi-stream Statistical N-gram Modeling with Application to Automatic Language Identification. In *INTERSPEECH*, number 1, pages 803–806.

Peter Ladefoged. 2001. *A Course in Phonetics*.volume 53. Cengage learning.

Lori F. Lamel and Jean-Luc Gauvain. 1994. Language Identification Using Phone-Based Acoustic Likelihoods. In *Proceedings of ICASSP '94. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume i, page I/293–I/296 vol.1.

Niels Landwehr, Mark Hall, and Eibe Frank. 2005. Logistic Model Trees. *Machine Learning*, 59(1-2):161–205.

Haizhou Li Haizhou Li, Bin Ma Bin Ma, and Chin-Hui Lee Chin-Hui Lee. 2007. A Vector Space Modeling Approach to Spoken Language Identification. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(1):271–284.

Richard P. Lippmann. 1997. Speech Recognition by Machines and Humans. *Speech Communication*, 22(1):1–15.

Ignacio Lopez-Moreno, Javier Gonzalez-Dominguez, Oldrich Plchot, David Martínez, Joaquin Gonzalez-Rodriguez, and Pedro Moreno. 2014. Automatic Language Identification Using Deep Neural Networks. *Icassp*:0–4.

David Martinez, Eduardo Lleida, Alfonso Ortega, and Antonio Miguel. 2013. Prosodic Features and Formant Modeling for an Ivector-based Language Recognition System. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 6847–6851. IEEE.

David Martínez, Oldřich Plchot, Lukáš Burget, Ondřej Glembek, and Pavel Matějka. 2011. Language Recognition in iVectors Space. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*(August):861–864.

Pavel Matejka, Petr Schwarz, Jan Cernocký, and Pavel Chytil. 2005. Tuning Phonotactic Language Identificaion System. Technical Report 4.

Yeshwant K. Muthusamy, Etienne Barnard, and Ronald a. Cole. 1994. Reviewing Automatic Language Identification. *IEEE Signal Processing Magazine*, 11(4):33–41.

Yeshwant Kumar Muthusamy, Kay M Berkling, T Arai, Ronald a Cole, and E Barnard. 1993. A Comparison of Approaches to Automatic Language Identification Using Telephone Speech. In *3rd European Conference on Speech Communication and Technology*, volume 2, pages 1307–1310.

Thangavelu Nagarajan and H. A. Murthy. 2004. Language identification using parallel syllable-like unit recognition. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–401. IEEE.

François Pellegrino and Régine Andre-Obrecht. 2000. Automatic language identification: an alternative approach to phonetic modelling. *Signal Processing*, 80(7):1231–1244.

John C. Platt. 1998. Sequential Minimal Optimization: a Fast Algorithm for Training Support Vector Machines. *Advances in Kernel MethodsSupport Vector Learning*, 208:1–21.

Olivier Pourret. 2008. *Bayesian Networks: a Practical Guide to Applications*.volume 73. John Wiley & Sons.

John Ross Quinlan. 1993. *Programs for Machine Learning*.volume 240. Elsevier.

John Ross Quinlan. 2014. *C4. 5: Programs for Machine Learning*. Elsevier.

Lawrence R. Rabiner. 1989. Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286.

Itahashi Shuichi and Du Liang. 1995. Language Identification Based on Speech Fundamental Frequency. In *4th European Conference on Speech Communication and Technology*, volume 2, pages 1359–1362.

Stanley S. Stevens. 1937. A Scale for the Measurement of the Psychological Magnitude Pitch. *The Journal of the Acoustical Society of America*, 8(3):185.

Stanley S. Stevens. 1939. The Relation of Pitch to the Duration of a Tone. *The Journal of the Acoustical Society of America*, 10(3):255.

Marc Sumner, Eibe Frank, and Mark Hall. 2005. Speeding up Logistic Model Tree Induction. In *Knowledge Discovery in Databases: PKDD 2005*, volume 3721, pages 675–683. Springer.

Ann E. Thyme-Gobbel and S. E. Hutchins. 1996. On Using Prosodic Cues in Automatic Language Identification. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP '96*, volume 3, pages 1768–1772.

Ingo R. Titze and Daniel W. Martin. 1998. Principles of Voice Production. *The Journal of the Acoustical Society of America*, 104(3):1148.

Pedro A. Torres-Carrasquillo, Douglas A. Reynolds, and J. R. Deller. 2002. Language Identification Using Gaussian Mixture Model Tokenization. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–757–I–760.

Geoffrey I. Webb. 2000. MultiBoosting: a Technique for Combining Boosting and Wagging. *Machine Learning*, 40(2):159–196.

Yiming Yang and Jan O Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, volume 97, pages 412–420.

Yeshwant K. Muthusamy, Ronald A. Cole, and Beatrice T. Oshika. 1992. The OGI Multi-language Telephone Speech Corpus. In *Proceedings of the International Conference on Spoken Language Proceedings*

(ICSLP, 現 INTERSPEECH), volume 92, pages 895–898. Citeseer.

Steve Young, Gunnar Evermann, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason, Valtcho Valtchev, and Phil Woodland. 2002. *The HTK book*.volume 3. Entropic Cambridge Research Laboratory Cambridge.

Stephen A Zahorian and Hongbing Hu. 2008a. YAAPT Pitch Tracking MATLAB Function. *The Journal of the Acoustical Society of America*, 123:4559–4571.

Stephen A. Zahorian and Hongbing Hu. 2008b. A Spectral/Temporal Method for Robust Fundamental Frequency Tracking. *The Journal of the Acoustical Society of America*, 123(6):4559–4571.

Marc A. Zissman. 1996. Comparison of Four Approaches to Automatic Language Identification of Telephone Speech. *IEEE Transactions on Speech and Audio Processing*, 4(1):31–44, January.

Marc A. Zissman and E. Singer. 1994. Automatic Language Identification of Telephone Speech Messages using Phoneme Recognition and N-gram Modeling. In *Proceedings of ICASSP '94. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume i, pages I–305.