Speaker Identification Using Semi-supervised Learning

Nikos Fazakis(✉), Stamatis Karlos, Sotiris Kotsiantis, and Kyriakos Sgarbas

University of Patras, Patras, Greece
fazakis@ece.upatras.gr, {stkarlos,sgarbas}@upatras.gr,
sotos@math.upatras.gr

Abstract. Semi-supervised classification methods use available unlabeled data, along with a small set of labeled examples, to increase the classification accuracy in comparison with training a supervised method using only the labeled data. In this work, a new semi-supervised method for speaker identification is presented. We present a comparison with other well-known semi-supervised and supervised classification methods on benchmark datasets and verify that the presented technique exhibits better accuracy in most cases.

Keywords: Semi-supervised learning · Speaker identification · Classification using labeled · Unlabeled data

1 Introduction

Labeled examples are often costly and time consuming to obtain, since labeling examples requires the effort of a human expert. On the other hand, unlabeled data is relatively easy to obtain in a number of domains. Semi-supervised classification methods use the available unlabeled data, along with a small set of labeled instances, to reduce the error rate in comparison with training a supervised classifier using only the labeled data [22]. To the best of our knowledge, there is no study that examines the efficiency of semi-supervised learning techniques in speaker identification that uses as base learners support vector machines and local based models.

The most known models for extracting useful characteristics for speech recognition are the source-filter model, which lead to extraction of Mel-frequency Cepstral coefficients (MFCC), Linear Predictive Codes (LPC), Perceptual Linear Prediction (PLP), PLP-Relative Spectra (PLP-RASTA) [2]. The reason why such various sets of features exist, is that Digital Speech Processing can be performed at three different levels so as to parameterize the speech. The first one examines the anatomy of human auditory system and tries to adjust its features to the average physical model of this. The second one considers speech phonemes, which constitute the basic component of speech, and the last one is associated with the linguistic nature of speech [14]. However, because of the

A. Ronzhin et al. (Eds.): SPECOM 2015, LNAI 9319, pp. 389–396, 2015.
DOI: 10.1007/978-3-319-23132-7_48

non-linear behavior of speech, there is a need for converting the field of frequency into another one, which may fit to human ear scale in a better way and can exploit the frequency domain features of speech. Consequently, the MFCCs features have been proved more efficient for this concept [32].

Grimaldi and Cummins [2] presented an experimental evaluation of different MFCC features for use in speaker identification. Those features were produced using speech data provided by the chains corpus, in a closed-set speaker identification task. The same wav files are used in our work. In this work, a new semi-supervised method for speaker identification is presented. We performed a comparison with other well-known semi-supervised and supervised classification methods and the presented technique had best accuracy in the tested data.

2 Speaker Identification Using Machine Learning

Mel-frequency Cepstral coefficients (MFCC) are popular features extracted from speech data for speaker identification. The speech signal is fragmented into frames and the MFCC features extracted from each frame show some temporal redundancy which forms the basis of fuzzy nearest neighbor classifier proposed in [19]. Khaled [1] used techniques of wavelet transform (WT) and neural network for speech based text-independent speaker identification. Lan et al. [6] examined extreme learning machine (ELM) on the text-independent speaker verification task and compared with SVM classifier. Empirical results showed that ELM classifiers performed better than SVM classifiers.

Pal et al. [28] illustrated, with the help of a bilingual speech corpus, how the well-known principal component transformation, in conjunction with the principle of classifier combination can be used to enhance the performance of the MFCC-GMM speaker recognition systems. Conventional speaker Identification systems use Gaussian mixture models and support vector machines (SVM) to model a speakers voice based on the speakers acoustic characteristics. Whereas GMMs needs more data to perform adequately and is computationally inexpensive, SVM on the other hand can do well with less data and is computationally expensive. Bourouba et al. [27] proposed a novel approach that combines the power of generative GMMs and discriminative support vector machines.

Dileep et al. [4] proposed to use the pyramid match kernel (PMK) based SVM classifier for speaker identification from the speech signal of an utterance represented as a set of local feature vectors. The main issue in building the PMK-based SVM classifier is the construction of a pyramid of histograms. Results of their studies show that the dynamic kernel SVM-based approaches give better performance than the state-of-the-art GMM-based approaches. Manikandan and Venkataramani [3] used modified One against All Support Vector Machine (SVM) classifier for speaker identification.

3 Semi-supervised Techniques

Sun [15] reviews theories developed to understand the properties of multi-view learning and gives a taxonomy of approaches according to the supervised and semi-supervised machine learning mechanisms involved. Self-training is a wrapper method usually used for semi-supervised classification [2]. In this process a classifier is first trained using the small set of labeled examples. Then unlabeled examples are classified using the trained learner. The classified unlabeled examples, for which the learner is high confident about its prediction (e.g. the first instances after the ranking of class probability values), are added to the training set along with their predicted class labels. In this way, the amount of training data increases due to the inclusion of the high-confidence unlabeled examples in the training set. Re-training of the classifier is done using the new enlarged training set and this process is repeated a fixed number of iterations until stopping criteria to be satisfied.

Co-Training is based on the assumption that the attribute space can be split into two disjoint subsets, and that each subset can produce correct classification [8]. Thus, a single learner is trained on each subset. Initially, both learners are trained only on labeled data. Then each learner is asked to classify a small number of unlabeled instances and the most confident predictions of each one learner are added to the training set of the other one. This procedure re-iterates for a number of times until a stopping criteria to be satisfied. Didaci et al. [21] evaluated co-training performance as a function of the size of the labeled training set. Results on real data sets, showed that co-training performance seems not be affected a lot by the training set size. On the other hand, Du et al. [25] made a number of experiments and concluded that based on small labeled training sets, verifying the sufficiency and independence assumptions or splitting single view into two views are unreliable.

Jiang et al. [20] proposed a co-training style algorithm which employs Naive Bayes and Support Vector Machine as base learners. The final prediction is given by the combination of base learners. Wang et al. [9] proposed to combine the probabilities of class membership with a distance metric between unlabeled instances and labeled instances. If two instances have the same class probability value, the one with the smaller distance will have larger chance to be selected.

Li and Zhou [11] proposed Co-Forest algorithm. According to this algorithm, a number of Random Trees are trained on bootstrap sample data from the data set. Then each Random Tree is refined with a small number of unlabeled instances during the training process and the final prediction is produced by majority voting. Deng and Guo [16] proposed a new Co-Forest algorithm named ADE-Co-Forest [7] which uses a data editing technique to identify and discard probably mislabeled instances during the iterations. RASCO [17] uses random attribute splits in order to train different learners. The unlabeled data are labeled and added to the training set based on the combination of decisions of the learners trained on different attribute splits. Tri-training algorithm has been proposed by [7]. In each round of tri-training algorithm, an unlabeled instance is labeled for a learner if the other two learners agree on the labeling.

Democratic co-learning [13] also uses multiple classifiers. Initially, each classifier is trained with the same data. The classifiers are then used to label the unlabeled data. Each instance is then labeled with the majority voting, and the labeled instance is added to the training set of the classifier whose prediction disagree with the majority.

4 Proposed Algorithm

The proposed method begins with a transformation of the speech signal to the feature space model in order to apply semi-supervised machine learning techniques. To be more specific, the procedure of extracting the MFCCs is based on a short-term spectral analysis method, in which speech signals are divided into short frames using mainly the Hamming window of length equal to either 1024 points or even less for less stationary signals, or bigger ones for the rest. Also, the choice of 50 % overlap between consecutive frames, seems to satisfy the majority of the different scenarios. Furthermore, the calculation of these parameters includes the computation of Fast Fourier Transform (FFT) of all the windowed speech segments. Then, the logarithmic Mel-scaled filter bank is applied to each one. The main characteristic of this scaling is that it combines both linearly spaced filter bank for frequencies lower than 1kHz and logarithmically spaced one for higher frequencies, without the temporal resolution in every frequency band being affected. The output of this stage is the mel spectrum coefficients which are strictly real numbers. Finally, Discrete Cosine Transformation (DCT) of any filter bank is performed during the last phase, computing the desired amount of MFCC coefficients for every frame. It is necessary to refer that in the most automatic speech recognition systems, the 0th coefficient of the MFCC cepstrum is ignored because of its unreliability [2]. This assumption will be supported during our experiments in this work. Also, the values of Min and Max Frequency that are inserted in MFCC extraction procedure, have been set to 0 Hz and 4 kHz, in order to cover the whole spectrum of speech signals. Self-training models do not make any specific assumption for the training data, but they accept that their own high-confident predictions are correct. However, it can lead to wrong predictions if noisy instances are classified as the most confident instances and merged into the training set. Of course, self-training will also fail if the small number of labeled examples cannot at all represent the underlying structure of the space, because the initial trained learner will produce bad predictions for the unlabeled data.

Most often speaker Identification systems use support vector machines (SVM) to model a speakers voice based on the speakers acoustic characteristics. SVMs [18] revolve around the notion of a "margin" - either side of a hyperplane that separates two data classes. Self-training cannot straightforward be applied to support vector machines. The confident examples are not too informative since most of them would have large distance from the decision boundary.

Naive Bayes classifier [24] is among the most popular learners used in the machine learning community. In this work, we combine the power of Naive Bayes and instance base learners. Combining instance-based learning with Naive Bayes is motivated by improving Naive Bayes through relaxing the conditional independence assumption using lazy learning. It is expected that there are no strong

```
Input: An initial set of labeled instances L and a set of unlabeled
instances U

Initialization:
 1) Initialize a shared training set EL by initial set of labeled instances
 2) Initialize a Support Vector Machines (SVM) classifier
 3) Initialize a Logistic Regression classifier
For a number of iterations do:
 4) Find the k(=100) nearest neighbors in EL using the selected distance
 metric (Euclidean in our implementation). Using as training instances
 the 100 instances train the simple Bayes classifier. Use local simple Bayes
 classifier to give the probabilities for each instance in U
 5) Use SVM classifier to give the probabilities for each instance in U
 6) Use Logistic Regression classifier to give the probabilities for each
 instance in U
 7) Average the probabilities of the three classifier and select the
 instances with the most confident predictions, remove them from U and
 add them to EL. In each about 1-2 instances per class are removed
 from U and added to EL

Output: Built the same ensemble of classifiers in the final labeled set
to predict the class labels of the test cases.
```

Fig. 1. The SelfSSL algorithm

dependences within the k nearest neighbors of the test instance, although the attribute dependences might be strong in the whole data [24]. Essentially, they are looking for a sub-space of the instance space in which the conditional independence assumption is true or almost true. Logistic regression [10] measures the relationship between the categorical dependent variable and one or more independent variables, which are usually continuous, by estimating probabilities. Logistic regression is not as accurate method as SVMs but exports more reliable probabilities for each instance classification.

Finally, the proposed algorithm (SelfSLL) is presented in Fig. 1. Combining the power of SVMs, Local Naive Bayes and Logistic Regression, the model predicts more accurate the class probability values. As a result, a number of most confident predictions of unlabeled instances can be added into the training set and the ensemble is retrained. The process is repeated until a stopping criterion is met.

For the implementation, it must be mentioned that we made use of the free available code of WEKA [22] and KEEL [31].

5 Experiments

The experiments are based on datasets extracted from the CHAINS Corpus (http://chains.ucd.ie/). The dataset consists of 16 different speakers who read 33 different sentences at a comfortable rate. In order to study the influence of the amount of labeled data, we take two different ratios when dividing the training set: 20 % for 8 speakers problem and 40 % for 16 speakers problem. These datasets have been partitioned using the 10-fold cross-validation procedure. For each generated fold, a given algorithm is trained with the examples contained in the rest of folds (training partition) and then tested with the current fold. It is noteworthy that test partitions are kept aside to evaluate the performance of the learning algorithm. Each training partition was divided into two parts: labeled and unlabeled examples. For the experiments, the proposed method has been compared with other state of the art algorithms integrated into the KEEL (Knowledge Extraction based on Evolutionary Learning) tool http://sci2s.ugr.es/keel/ [31]. For the tested algorithms the default parameters of KEEL and WEKA have been used. The classification accuracy of each supervised and semi-supervised learning algorithm tested in our study is presented in Tables 1 and 2 respectively.

The proposed method performs better than the tested state of the art algorithms. The presented approach can utilize automatically labeled data to augment a smaller, manually labeled dataset and thus improve the performance.

Table 1. Accuracy of each tested supervised learning method.

Algorithms	20 % Instances of 8 speakers	40 % Instances of 16 speakers
SupervisedNN	0.6401	0.5875
SupervisedNB	0.6997	0.6032
SupervisedC45	0.4561	0.3467
SupervisedSMO	0.8001	0.7685
SupervisedSLL	0.7968	0.7696
SupervisedLogistic	0.6921	0.6877
SupervisedLNB	0.7433	0.6942

Table 2. Accuracy of each tested semi-supervised learning method.

Algorithms	20 % Instances of 8 speakers	40 % instances of 16 speakers
SelftrainNN	0.6233	0.5718
SelftrainNB	0.6004	0.5354
SelftrainC45	0.4399	0.3639
SelftrainSMO	0.7808	0.7455
SelfSLL	0.8145	0.7819
TriTrainC45NBNN	0.6569	0.5415
CoTrainNNC45NN	0.6348	0.5844
CoForest	0.5553	0.4657
Rasco	0.2712	0.2821

6 Conclusion

In this work, a new semi-supervised method for speaker identification was presented. We performed a comparison with other well-known semi-supervised classification methods on standard benchmark datasets and the presented technique had the best accuracy in the specific datasets. Due to the encouraging results obtained from these experiments, we can expect that the proposed technique can be effectively applied to the classification task in the real world case giving slightly better accuracy than the traditional semi-supervised approaches. In spite of these results, no general method will work always.

References

1. Khaled, D.: Wavelet entropy and neural network for text-independent speaker identification. Engg. Appl. Artif. Intell. **24**, 796–802 (2011)
2. Grimaldi, M., Cummins, F.: Speaker identification using instantaneous frequencies. IEEE TASLP **16**(6), 1097–1111 (2008)
3. Manikandan, J., Venkataramani, B.: Design of a real time automatic speech recognition system using modified one against all SVM classifier. Microproc. Microsyst. **35**(6), 568–578 (2011)
4. Dileep, A., Chandra, C.: Speaker recognition using pyramid match kernel based support vector machines. Int. J. Speech Technol. **15**(3), 365–379 (2012)
5. Friedhelm, S., Edmondo, T.: Pattern classification and clustering: a review of partially supervised learning approaches. Pattern Recogn. Lett. **37**, 4–14 (2014)
6. Lan, Y., Hu, Z., Soh, Y.C., Huang, G.-B.: An extreme learning machine approach for speaker recognition. Neural Comput. Appl. **22**(3–4), 417–425 (2013)
7. Zhi-Hua, Z., Li, M.: Tri-training: exploiting unlabeled data using three classifiers. IEEE TKDE **17**(11), 1529–1541 (2005)
8. Chapelle, O., Schlkopf, B., Zien, A.: Semi-supervised learning. MIT Press, Cambridge (2006)
9. Shuang, W., Linsheng, W., Licheng, J., Hongying, L.: Improve the performance of co-training by committee with refinement of class probability estimations. Neurocomputing **136**, 30–40 (2014)
10. Xu, J., He, H., Man, H.: DCPE co-training for classification. Neurocomputing **86**, 75–85 (2012)
11. Li, M., Zhou, Z.: Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. IEEE TSMC **37**, 1088–1098 (2007)
12. Hady, M., Schwenker, F.: Co-training by committee: a new semi-supervised learning framework, In: Proceedings of the IEEE International Conference on Data Mining Workshops, pp. 563–572 (2008)
13. Zhou, Y., Goldman, S.: Democratic co-learning. In: 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'04), pp. 594–202 (2004)
14. Hyon, S., Dang, J., Feng, H., Wang, H., Honda, K.: Detection of speaker individual information using a phoneme effect suppression method. Speech Commun. **57**, 87–100 (2014)
15. Sun, S.: A survey of multi-view machine learning. Neural Comput. Appl. **23**(7–8), 2031–2038 (2013)
16. Deng, C., Guo, M.Z.: A new co-training-style random forest for computer aided diagnosis. J. Intell. Inf. Syst. **36**, 253–281 (2011)

17. Wang, J., Luo, S., Zeng, X.: A random subspace method for co-training. In: IEEE International Joint Conference on Computational Intelligence, pp. 195–200 (2008)
18. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press, Cambridge (2000)
19. Susan, S., Sharma, S.: A fuzzy nearest neighbor classifier for speaker identification. In: 4th International Conference on Computational Intelligence and Communication Networks, CICN 2012, pp. 842–845 (2012)
20. Jiang, Z., Zhang, S., Zeng, J.: A hybrid generative/discriminative method for semi-supervised classification. Knowl.-Based Syst. **37**, 137–145 (2013)
21. Didaci, L., Fumera, G., Roli, F.: Analysis of co-training algorithm with very small training sets. In: Gimel'farb, G., Hancock, E., Imiya, A., Kuijper, A., Kudo, M., Omachi, S., Windeatt, T., Yamada, K. (eds.) SSPR &SPR 2012, vol. 7626, pp. 719–726. Springer, Berlin (2012)
22. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.: The WEKA data mining software: an update. SIGKDD Explor. **11**(1), 10–18 (2009)
23. Frank, E., Hall, M., Pfahringer, B.: Locally weighted naive Bayes. In: 19th Conference on Uncertainty in Artificial Intelligence. Mexico (2003)
24. Domingos, P., Pazzani, M.: On the optimality of the simple Bayesian classifier under zero-one loss. Mach. Learn. **29**, 103–130 (1997)
25. Du, J., Ling, C.X., Zhou, Z.-H.: When does cotraining work in real data? IEEE TKDE **23**(5), 788–799 (2011)
26. Goldberg, X.: Introduction to semi-supervised learning. In: Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan Claypool (2009)
27. Bourouba, H., Korba, C.A., Djemili, R.: Novel approach in speaker identification using SVM and GMM. Control Engg. Appl. Inf. **15**(3), 87–95 (2013)
28. Pal, A., Bose, S., Basak, G.K., Mukhopadhyay, A.: Speaker identification by aggregating Gaussian mixture models (GMMs) based on uncorrelated MFCC-derived features. Int. J. Pattern Recogn. Artif. Intell. **28**(4), 25 (2014)
29. Zhao, X., Wang, Y., Wang, D.: Robust speaker identification in noisy and reverberant conditions. IEEE TASLP **22**(4), 836–845 (2014)
30. Alcal-Fdez, J., Fernandez, A., Luengo, J., Derrac, J., Garca, S., Snchez, L.: KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J. Multi.-Valued Logic Soft Comput. **17**(2–3), 255–287 (2011)
31. Triguero, I., Garca, S., Herrera, F.: Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. Knowl. Inf. Syst. **42**(2), 245–284 (2015)
32. Namrata, D.: Feature extraction methods LPC, PLP and MFCC in speech recognition. Int. J. Adv. Res. Engg Technol. **1**(6), 1–4 (2013)