SPEECH ACTIVITY DETECTION: AN ECONOMICS APPROACH

TJ Tsai, Nelson Morgan

University of California Berkeley, Berkeley, CA, USA
International Computer Science Institute, Berkeley, CA, USA

ABSTRACT

This paper proposes an approach to frame-level speech activity detection based on the extended metaphor of an economics marketplace. As in a real marketplace, the simulated marketplace encourages features to specialize. Features that might not have impressive average performance across the entire data set might nonetheless perform very well on a subset of the data, and the marketplace capitalizes on this specialization by consulting the features only when their expertise is relevant. On an experimental data set, we show that the framework is able to effectively utilize the expertise of a set of voicing-related features. For the 50% of the data that fell within these features' realm of expertise, we observe an 83% reduction in false alarm errors and 19% reduction in miss detect errors compared to a baseline HMM-GMM system with MFCCs. Even when we consult these features for the entire data set, thus including the other 50% of data outside their realm of expertise, we still observe a 20% total reduction in equal error rate compared to the baseline system. Analysis of the marketplace transactions also yields useful insight into how the errors are distributed across the data and which types of features are most useful.

Index Terms— speech activity detection, feature specialization

1. INTRODUCTION

This paper introduces a framework for speech activity detection (SAD) based on the metaphor of an economics marketplace. In this section we introduce the main concepts behind the framework at a high level and discuss the influences of previous work. There are three main concepts explored in this framework, and each of these three is discussed in turn.

The first main concept is that our framework tries to achieve a specific performance target. Most SAD systems are evaluated by sweeping across a range of one specific parameter to generate a receiver operation characteristic (ROC) or detection error tradeoff (DET) curve [1][2]. The ROC/DET curve describes the tradeoff between false alarm (FA) and miss detect (MD) errors. The point on the tradeoff curve closest to the desired performance goal is identified, and this operating point is chosen. In other words, the performance goal is not taken into consideration until after the system has already been evaluated. Several approaches deal with this issue by training the classifier using an error criterion based on some aspect of the ROC curve, such as the area under the curve [3], the detection accuracy for a given range of acceptable FA rates [4], or the performance near a specific point in the ROC plane [5]. The proposed approach is similar to [5] in attempting to achieve a specific performance target, but with the major difference that the performance target information is incorporated during the prediction phase rather than the training phase. During the prediction phase, this framework allocates a targeted maximum number of FA and MD errors and treats these "allowable" errors as valuable, constrained resources that can be traded in a marketplace. The performance target is thus taken into account from the beginning of the simulation, and predictions are strategically tailored to achieve the specific performance target to the best of the system's ability.

The second main concept is to treat each feature as a weak classifier and to assign weights to the weak classifiers in a dynamic fashion. This approach draws heavily from the Adaboost algorithm, which considers a set of weak classifiers and generates a prediction based on a weighted combination of the weak classifiers' predictions. In Adaboost, the weights assigned to the weak classifiers are determined during the training phase and are fixed during the prediction phase. [6] is the original paper by Freund and Schapire. Adaboost with trees was touted by Breiman [7] to be the "best off-the-shelf classifier in the world" and, indeed, we showed in [8] that Adaboost can be very effective for SAD in noisy environments when used with appropriate features. One major difference between our current approach and Adaboost is that, instead of having the weights of the classifiers be determined beforehand and fixed during prediction, we instead determine the weights dynamically by considering how confident each weak classifier is in its current prediction. Several different dynamic weighting schemes are explored in [9], [10], [11], and [12]. In this paper, we adopt the approach in [12] referred to as dynamic selection, where we assign all the weight to the most confident weak classifier and zero weight to all other weak classifiers. In other words, our strategy is to take each individual prediction problem before our committee, ask the question 'Who knows the answer to this question?' and let

the committee member with greatest confidence supply an answer. Because features are consulted only when their expertise is relevant, we refer to our 'experts' as specialist features. A more detailed explanation will be given in section 2.

The third main concept is to generate speech-nonspeech predictions in order of confidence rather than in order of time. In other words, rather than generating our predictions in sequential order in time, we instead generate predictions starting with the frame that we have the most confidence in and progress towards frames that we have less and less confidence in. So, for example, if we are trying to generate speech-nonspeech predictions on a file with 6000 frames, we might first generate a prediction for frame 5300, followed by a prediction for frame 500, and so on. Changing the order in which we generate predictions will probably not affect our accuracy significantly, but this reordering leads to interesting insights into the data that would not be available to us otherwise. [5] considers a similar reordering of data points based on the classifier's posterior for the purpose of re-weighting data samples in an iterative training process, but here we simply use the reordering during the prediction phase as a diagnostic tool to gain insight into the data and the features. As we will show later in this paper, this reordering allows us to understand how errors are distributed across the data and to identify which features are most useful for high accuracy predictions.

The rest of the paper is organized as follows. Section 2 describes the conceptual framework in detail. Section 3 explains the experimental set up. Section 4 shows results on the experimental data set and provides an analysis of the results. Section 5 summarizes the main ideas and concludes the work.

2. SYSTEM DESCRIPTION

There are two main components of this framework: training the models and running the economic marketplace simulation.

We first describe training the models. The models for each feature are computed independently and consist of two functions f and g. The function f is simply a histogram of the feature values. The function g uses the same histogram bins as f, but instead indicates the fraction of frames that are speech (i.e. an empirical speech posterior probability). In other words, if the value of g for a certain histogram bin is .2, this tells us that 20% of the frames that fall in that bin are speech. So, to train a model, we simply need to calculate the feature on all training frames, classify each frame into a histogram bin according to the feature value, and accumulate counts accordingly. Note that the models for each feature are completely independent of one another. The collection of functions f and g for all features comprises the training models.

Next, we turn our attention to the economic marketplace simulation. Figure 1 is a depiction of the marketplace that will help build a mental map in the reader's mind and serve as an explanation aid. The first row of circles at the top represent speech frames, and can be thought of as commodities

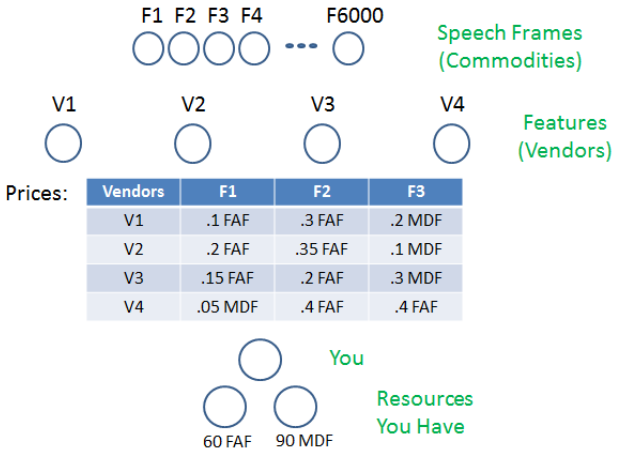

Fig. 1. Conceptual map of the economics marketplace. The goal is to purchase all the frames by carrying out transactions with the feature vendors. The table shows the prices that the vendors charge for the first 3 frames.

that we would like to purchase. The act of purchasing a frame corresponds to making a speech-nonspeech prediction on that frame. Our goal, then, is to eventually purchase all of the frames. The second row of circles represents features, and can be thought of as vendors through which one can purchase the commodities. For example, one feature vendor might be an energy feature, another feature vendor might be the first MFCC coefficient, and so on. The third row (below the table) contains a single circle which represents you, the consumer. Your goal in the simulation is to purchase all the commodities from the various vendors in the marketplace as cheaply as possible. The fourth row contains two circles which represent the two types of currencies that you can use as money to purchase the commodities. The currencies will be referred to as units of faf and mdf, which stand for false alarm frames and miss detect frames.

At this point it is helpful to point out a few observations about the marketplace simulation. First, as in a real marketplace, the vendors offer the commodities at different prices. Each element of this vendor-commodity price table is g(X) mdf or 1-g(X) faf. For example, if g(X)=.99, we can predict nonspeech and have a 99% probability of making a MD error (.99 mdf), or we can predict speech and have a 1% probability of making a FA error (.01 faf). Note, then, that for any given commodity, the vendor will offer a price in either faf or mdf. How do we compare prices in different currencies? This leads to our second observation, which is that the exchange rate between mdf and faf changes dynamically depending on how much faf and mdf you currently have. The reasoning here is that we value the different currencies according to how scarce that currency is. If we have equal amounts of faf and mdf, we have no preference on which type of mistake (FA or MD) we

make. However, if we have 100 faf and 1 mdf, we are very loathe to make a MD error. In our simulation, we simply used the ratio between the amount of faf and mdf currently owned as the exchange rate. Third, as noted previously, the amount of money we begin the simulation with is the estimated number of FA mistakes and MD mistakes we can make according to our performance target.

What is the best purchasing strategy of the consumer given his limited resources? The optimal strategy is obvious: purchase the cheapest frame that is available in the marketplace.[1] The transactions continue until the consumer has purchased all of the frames. What happens, though, if you run out of money? In this case, we can simply give ourselves more money to ensure that we eventually purchase all of the frames. The only ramification of giving ourselves more money is that it means we probably won't meet our performance targets.

3. EXPERIMENTAL SET UP

We ran simulations using data from the DARPA RATS program. The data consists of conversations recorded over various radio transmission links. In general, the audio data is very noisy and contains highly non-stationary noise, including high energy non-transmission regions. Due to ground truth label integrity issues, we randomly selected 1 minute segments and manually verified the labels, throwing out any segments that had poor labels. Our final data set consisted of 523 training segments and 324 evaluation segments. For more information on the data and on other SAD approaches proposed for this data set, see [13], [14], and [15].

The features we used in the simulations consist of a base voicing feature and a set of 220 derived voicing features.[2] The base feature is the probability of voicing as estimated by a subband autocorrelation pitch tracker, which is described in [17]. Using this probability of voicing at every frame as a base feature, we then derived a family of features by calculating statistics on windows of various sizes. The statistics we considered were the minimum, the maximum, and various quantiles in between (where, for example, the 50% quantile would correspond to the median). We considered windows up to 2 seconds long. In total, we had 221 voicing features.

4. RESULTS

The results of the simulations are shown in figure 2. The solid line is (what we will call) the error trajectory for the economics marketplace simulation using the 221 derived voicing features. An error trajectory begins at the (0,0) point in the ROC plane and shows the accumulation of FA and MD errors over the course of the simulation. Each plotted point along the trajectory demarcates 5% of the frames in each simulation (averaged across the evaluation data set). The error trajectory allows us to visualize how the errors in the system are distributed across the data, starting with the highest confidence frames and progressing towards the lowest confidence frames. We are particularly interested in the distribution of demarcated points close to the origin in the ROC plane, since this is the 'region of expertise' of these specialist features. What we would like to see is an extremely dense cluster of demarcated points close to the origin, for this would imply that the specialist features are very effective for a significant fraction of the data. The dotted line in figure 2 is an imputed error trajectory for a standard hidden markov model with gaussian mixture models (HMM-GMM) using 39 dimensional MFCCs. The 'imputed' term refers to the fact that we are scoring a different system but adopting the ordering of frames from our marketplace simulation. Comparing these two trajectories can tell us, for example, if frames that are 'easy' for the specialist features are also 'easy' for our reference baseline system.

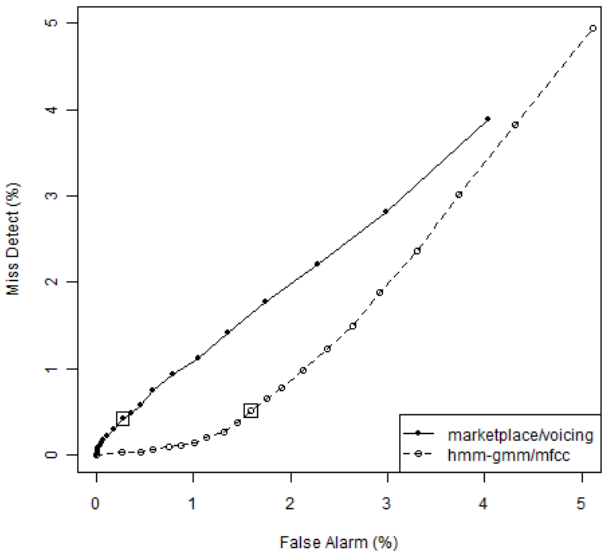

Fig. 2. Comparison of error trajectories at each system's equal error rate point. The solid trajectory shows the accumulation of errors as the marketplace simulation progresses. The dotted line is the imputed error trajectory for a baseline system where the ordering has been adopted from the marketplace simulation. Each segment represents 5% of the data. The boxed data points show the cumulative error rates for the first 50% of the data.

There are three things to notice about the results we observe in figure 2. First, the specialist features definitely spe-

[1] This statement should become quite obvious to any graduate student if you simply replace the word 'frame' with 'food'.

[2] Initial experiments were also carried out with spectrotemporal modulation features such as those explored in [8] and [16], but these features were much less effective than voicing features and so are not included in this work.

cialize. The spacing of the demarcated points in the solid line is very dense at the beginning of the trajectory and comparatively very sparse towards the end of the trajectory, indicating that the features perform much better on frames they are confident in. In fact, the 50% of the data that the specialist features are most confident in account for only 7% of the total FA errors and 11% of the total MD errors (this is denoted by the boxed point on the solid line). In contrast, the 13% of the data that the specialist features are least confident in account for 50% of the total number of both FA and MD errors (this corresponds roughly to the point at 2%FA and 2%MD). In other words, our specialist features are very smart in their realm of expertise and very dumb in their realm of ignorance. Second, the features' expertise level is very high. On the 50% of the data that the specialist features are most confident in, the specialist features have 83% fewer FA errors and 19% fewer MD errors compared to the baseline HMM-GMM system (this is indicated by the two boxed points). Within their realm of expertise, the specialist features' performance would be hard to beat. Third, both systems agree on what is hard. Note that both systems have roughly the same (poor) performance on the last few segments of their trajectories. This is an interesting coincidence given how different the two systems are both in their decoding algorithms and their features. It would be interesting to investigate what fraction of frames in these segments are located close to a speech-nonspeech boundary in the ground truth labels. It could be that the performance on these last few segments is affected more by the accuracy of ground truth annotations rather than by the effectiveness of the SAD system.

In addition to analyzing the error trajectory, we also looked at how many "sales" each feature vendor completed throughout the simulation. There were three notable observations from this sales analysis. First, the sales distribution resembled a Zipfian distribution. In other words, the top few vendors accounted for a significant fraction of sales, and there was a long tail in the distribution constituted of many vendors that only carried out a few sales. For example, when we consider the sales distribution on the 50% of the data within the features' realm of expertise, the top 7 vendors made up 50% of the sales while the worst 100 vendors made up less than 2% of the sales. Second, the feature vendors specialized individually. Earlier we made the point that the features specialize corporately as a group on a subset of the data, but here we note that the features also specialize individually in only predicting speech or only predicting nonspeech. On the first 50% of the data, 183 of the 221 feature vendors exclusively made only one type of prediction (either speech or nonspeech but not both), and these 183 vendors made up 99.5% of the sales. Apparently, in a very cut-throat marketplace the only way for a feature vendor to survive is to specialize in making one type of prediction. Third, there seems to be 2 different types of information that are especially useful for high-accuracy SAD predictions. Among the top 7 feature vendors on the first 50% of the data, 6 features were the maximum of voicing probability over long windows (ranging from 1.4 to 2.0 seconds). So, the first type of information is identifying long windows in which there is little or no voicing, which we can confidently predict as nonspeech. The other remaining feature (among the top 7) was the 40% quantile of voicing probability over a .5 second window. This second type of information is identifying regions with high average voicing probability over a medium time scale, which we can confidently predict as speech. Analysis of vendor sales is yet another aspect of the marketplace simulation that yields interesting insights into the effectiveness of the features.

5. CONCLUSION

We have introduced a framework to compute frame-level speech activity detection based on the extended metaphor of an economics marketplace. This framework has three main concepts. First, the framework attempts to achieve a specific performance target, and predictions are made to try to achieve the specified target to the best of the system's ability. Second, each feature is considered as a weak classifier, and the weak classifier with highest confidence determines the prediction on every frame. Third, predictions are generated in order of confidence rather than in a temporally sequential fashion. Based on simulations using DARPA RATS data, we showed that a set of voicing features specialize very well on a subset of the data. Within the 50% of the data that the features had most confidence in, we observed 83% fewer false alarm errors and 19% fewer miss detect errors compared to a baseline HMM-GMM system. Analysis of the transactions in the marketplace simulation also provides useful diagnostic information such as how errors are distributed across the data and which features are the most useful.

Future work includes (1) exploring alternative dynamic weighting schemes that incorporate multiple weak classifiers' predictions, (2) investigating other types of specialist features in addition to voicing features, (3) diagnosing discrepancies between the specified performance target and the actual performance results, (4) combining this framework with other systems so that the specialist features are utilized only when their expertise is relevant, and (5) analyzing transactions in the marketplace simulation for other purposes such as feature selection.

6. ACKNOWLEDGMENTS

This material is based on work supported by DARPA under contract no. D10PC20024. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of DARPA or its contracting agent, the US Department of the Interior, National Business Center, Acquisition & Property Management Division, Southwest Branch.

7. REFERENCES

[1] T. Fawcett, "Roc graphs: Notes and practical considerations for researchers," *Machine Learning*, vol. 31, pp. 1–38, 2004.

[2] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The det curve in assessment of detection task performance," Tech. Rep., DTIC Document, 1997.

[3] L. Yan, R. Dodier, M.C. Mozer, and R. Wolniewicz, "Optimizing classifier performance via the wilcoxon-mann-whitney statistics," in *Proceedings of the 20th international conference on machine learning*. Citeseer, 2003, pp. 848–855.

[4] E.I. Chang and R.P. Lippmann, "Figure of merit training for detection and spotting," *Advances in Neural Information Processing Systems*, pp. 1019–1019, 1994.

[5] M.C. Mozer, R. Dodier, M.D. Colagrosso, C. Guerra-Salcedo, and R. Wolniewicz, "Prodding the roc curve: Constrained optimization of classifier performance," in *Proceedings NIPS*, 2001, vol. 13.

[6] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Computational learning theory*. Springer, 1995, pp. 23–37.

[7] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[8] T. Tsai and N. Morgan, "Longer features: They do a speech detector good," in *Proceedings of Interspeech 2012, Portland, Oregon*, 2012.

[9] J.R. Quinlan, "Bagging, boosting, and c4. 5," in *Proceedings of the National Conference on Artificial Intelligence*, 1996, pp. 725–730.

[10] P. Moerland and E. Mayoraz, "Dynaboost: Combining boosted hypotheses in a dynamic way," *IDIAP-RR, Switzerland*, 1999.

[11] R. Valdovinos and J. Sánchez, "Combining multiple classifiers with dynamic weighted voting," *Hybrid Artificial Intelligence Systems*, pp. 510–516, 2009.

[12] A. Tsymbal and S. Puuronen, "Bagging and boosting with dynamic integration of classifiers," *Principles of Data Mining and Knowledge Discovery*, pp. 195–206, 2000.

[13] K. Walker and S. Strassel, "The rats radio traffic collection system," in *Odyssey 2012-The Speaker and Language Recognition Workshop*, 2012.

[14] T. Ng, B. Zhang, L. Nguyen, S. Matsoukas, K. Vesely, P. Matejka, X. Zhu, and N. Mesgarani, "Developing a speech activity detection system for the darpa rats program," in *Proceedings of Interspeech 2012, Portland, Oregon*, 2012.

[15] S. Thomas, S.H. Mallidi, T. Janu, H. Hermansky, N. Mesgarani, X. Zhou, S. Shamma, T. Ng, B. Zhang, L. Nguyen, et al., "Acoustic and data-driven features for robust speech activity detection," in *Proceedings of Interspeech 2012, Portland, Oregon*, 2012.

[16] B.T. Meyer, S.V. Ravuri, M.R. Schädler, and N. Morgan, "Comparing different flavors of spectro-temporal features for asr," in *Proceedings of Interspeech*, 2011, pp. 1269–1272.

[17] B.S. Lee and D. Ellis, "Noise robust pitch tracking by subband autocorrelation classification," in *Proceedings of Interspeech 2012, Portland, Oregon*, 2012.