### SYSTEM DESCRIPTION FOR TL@NTU

Zhi Hao Lim, Wei Rao

Qing Wang

# Temasek Laboratories@NTU Department Nanyang Technological University Singapore

School of Computer Science Northwestern Polytechnical University Xian, China

#### **ABSTRACT**

This is a short description about the system for team TL@NTU. This challenge involves 2 tasks, namely 1) Top-1 and 2) Top-S speaker detection. Top-S blacklist detection can be interpreted as a binary classification task, hence the team decided to use the Random Forest ensemble approach to build a binary classifier. The Top-1 task can be interpreted as a multiclass classification task, the team has decided to use a modified cosine distance scoring approach.

*Index Terms*— Random Forest, Cosine Distance Scoring, Ridge Regression

### 1. INTRODUCTION

The MCE 2018 is the 1st Multi-target speaker detection and identification Challenge Evaluation. Organizer provided ivector dataset and baseline code, so all participants could start from the same baseline [1].

For the fixed condition primary submission, there are 2 task, 1) Top-1 and 2) Top-S speaker detection. For the first task, it can be interpreted as a multiclass classification with high number of classes. The problem is as such: "If an utterance is spoken by a speaker from the blacklist, who is it?". This task requires us to search through the list of blacklisted speakers and find the most likely one. The second problem is more straightforward, in the sense that it can be interpreted as a binary classification task. The problem is as such: "Given an utterance, is it from the blacklist?"

However, it is not as straightforward as it seems and one should be careful as each task has its own challenges. In the first task, the amount of classes is over 3000, which makes it a highly multiclass classification task. [2] shown that with the increased number of classes, increases the inter-class confusabilities. Hence the classification tasks with such high number of classes is often very challenging. Furthermore, under the total variability framework [3], ivectors extracted contains both channels and speaker variabilities. In other words, the features provided may not contain purely speaker discriminative information. One final point to note is that in real life situations, it is reasonable to assume that the blacklist

changes over time. This means that the speakers are being randomly allocated a binary class label, which in turn implies that the blacklist space maybe highly non-linear.

With such complexities in both the tasks, the team has decided to use a more stochastic approach to tackle the Top-S task and a more tried and tested approach of using cosine distance scoring for the Top-1 task. The motivation behind for using Random Forests is because the lack clarity on the structure of data provided. There are many ways to explore the structure of a data [4], this includes the classical techniques like ANOVA or Factor Analysis. However as hypothesize by the Team, the data may not contain speaker discriminative information or the blacklist space contain alot of overlapping regions with the whitelist space. This is where classical approaches may fail and maybe better off using a stochastic (random) approach for dimension reduction and classification instead, most which are explained in [5].

The rest of the paper would contain the 2 system descriptions of the Team's approaches, Random Forest for Top-S detection and Cosine Distance for Top-1 detection.

## 2. SYSTEM DESCRIPTION OF RANDOM FOREST ENSEMBLE FOR TOP-S DETECTION

As mentioned before, this task is interpreted as a binary classification task so for the Top-S detection task, the Team used an ensemble approach. Further, inspired by [6], the Team apply a random rotation to make the weak learners more diverse. For explanation purposes, the blacklist is class 1 and whitelist is class 0. The algorithm are all implemented in *Matlab*. For a single random tree, the algorithm is as follows:

- 1. Randomly choose 4096 points from each class (aka stratified sampling) in the **training** set
- 2. Project randomly from 600 dimensions to 64 dimensions (via random subspace selection [7])
- 3. Apply a random rotation to the projected data (using the *sprandsym* function available in *Matlab*)
- 4. Train a binary decision tree (using the *fitctree* function available in *Matlab*) with NO pruning.

Then repeating above, we create an ensemble of 4096 decision trees (weak learners). Many ensemble pruning approaches were explored in [8], however the team simply choose to iterate through and select the combination with lowest error on the **development** set. The final number of decision trees are 3793.

## 3. SYSTEM DESCRIPTION OF MODIFIED COSINE DISTANCE SCORING FOR TOP-1 DETECTION

The Top-1 detection task is interpreted as a multiclass classification task. Inspired by the explanation in [9], the Team decided to use a simple one-versus-all weighted ridge regression approach for this task. The algorithm is as follows:

- 1. Let  $X \in \mathbb{R}^{n \times d}$  be the blacklist training set, where n = 10893 and d = 600
- Perform length normalization for each row (aka each blacklist utterance) and center the data by subtracting the sample mean
  - $\frac{x_i}{||x_i||} \hat{\mu} \to x_i$  where  $\hat{\mu} = \frac{1}{n} \sum_{i=1}^n x_i$
- 3. Then add the bias term to X for regression:
  - $[X, \mathbf{1}_n] \to X$  where  $\mathbf{1}_n \in \mathbb{R}^n$
- 4. For each class  $i = 1 \dots C$  where C = 3631 in this case, construct the binary vector  $y_i \in \{0,1\}^n$  as such:
  - $y_{ij}=1$  if the sample index j is in class i where  $j=1\dots n$  and  $y_{ij}=0$  otherwise
- 5. For each class i, generate the diagonal weight matrix  $W_i \in \mathbb{R}^{n \times n}$  as follows
  - $W_{jj}=w$  if the sample index j is in class i and  $W_{jj}=1$  otherwise. For this task, we set w=40
- 6. For each class i, evaluate
  - $\beta_i = (X^TW_iX + \alpha I)^{-1}X^Ty_i$  where  $\alpha = 5$  in this case and  $I \in R^{(d+1)\times (d+1)}$

Finally, obtain the  $\beta=[\beta_1,\dots,\beta_C]\in R^{(d+1)\times C}$  scoring matrix. For testing let  $Z\in R^{m\times d}$  be the incoming development set, which in this case means m=3631 and d=600. First perform the same pre-processing steps (2) and (3) on Z, namely

$$\frac{z_i}{||z_i||} - \hat{\mu} \to z_i \ \ \text{and} \ \ [Z, \mathbf{1}_m] \to Z$$

Where  $\hat{\mu} = \frac{1}{m} \sum_{i=1}^{m} z_i$ . Then score via

 $Z\beta$ 

and take the maximum row-wise. This is equivalent to:

$$Class(z_i) = \arg\max_{k} \left\{ z_i^T \beta_k \right\}$$

For each sample  $z_i$  in the development set.

#### 4. RESULTS ON DEVELOPMENT SET

The results are presented in Table 4. The proposed approaches have shown to be useful in both the classification task.

	Baseline (EER)	Proposed (EER)	Confusion error
Top-S	2.00	13.41	492
Top-1	1.24	10.72	369

**Table 1**. EERs of baseline and proposed

### 5. CONCLUSION AND FUTURE WORKS

The Team has completed both the classification tasks with some success. However, the analysis is not complete and the TL@NTU Team will submit another paper to ICASSP 2019 to further describe the Team's findings on the database provided by the organizers of MCE2018. Findings include effects of using artificial data [10] and very sparse random projections [11] in this challenge.

### 6. REFERENCES

- [1] Suwon Shon, Najim Dehak, Douglas Reynolds, and James Glass, "Mce 2018: The 1st multi-target speaker detection and identification challenge evaluation (mce) plan, dataset and baseline system," *ArXiv e-prints arXiv:1807.06663*, 2018.
- [2] Maya R. Gupta, Samy Bengio, and Jason Weston, "Training highly multiclass classifiers," *Journal of Machine Learning Research*, vol. 15, pp. 1461–1492, 2014.
- [3] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011
- [4] David C. Hoaglin, "John w. tukey and data analysis," *Statistical Science*, vol. 18, no. 3, pp. 311–318, 2003.
- [5] Maryam Sabzevari, Gonzalo Martínez-Muñoz, and Alberto Suárez, "Vote-boosting ensembles," *CoRR*, vol. abs/1606.09458, 2016.
- [6] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation forest: A new classifier ensemble method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1619–1630, Oct 2006.

- [7] Tin Kam Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, Aug 1998.
- [8] G. Martnez-Muoz, D. Hernndez-Lobato, and A. Surez, "An analysis of ensemble pruning techniques based on ordered aggregation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 245–259, Feb 2009.
- [9] Ryan Rifkin and Aldebaro Klautau, "In defense of onevs-all classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, Dec. 2004.
- [10] Prem Melville, *Creating Diverse Ensemble Classifiers to Reduce Supervision*, Ph.D. thesis, Department of Computer Sciences, University of Texas at Austin, November 2005, Technical Report TR-05-49.
- [11] Ping Li, Trevor J. Hastie, and Kenneth W. Church, "Very sparse random projections," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2006, KDD '06, pp. 287–296, ACM.