1. INTRODUCTION

This paper briefly describes the language recognition systems developed by the Software Technology Working Group (http://gtts.ehu.es) of the University of the Basque Country (EHU) for the NIST 2017 Language Recognition Evaluation. The submitted system uses the Brno University of Technology (BUT) 80 dimension bottleneck features [1] trained on FisherEnglish (2423 triphones) and follows the Total Variability Factor Analysis (i-Vector) approach [2]. The i-Vector extractor (1024 Gaussians and 400 dimensional i-Vector) is based on the Sidekit Toolkit [3] and it is followed by a Gaussian Linear Classifier and a Discriminative Gaussian Back-end. Linear logistic regression calibration is applied to the final scores using the FoCal Toolkit [4].

2. DATASETS

The meta-information of the audio files was not used to create the datasets. That is, each dataset contains audio files with different lengths, formats and sources. The data was partitioned as follows:

- **Train** (16201 files): All available data from LDC2017E22 (2017 NIST LRE Training Data). This Dataset was randomly reduced to create another dataset:
  - **TrainBalanced** (4566 files): Random subset of files summing around 8000 seconds of voiced feature vectors per target language.

- **Dev** (3659 files): All available data from LDC2017E23 (2017 NIST LRE Development Data). The audio files lre17_ytgfwvpa.flac and lre17_gpupyoiu.flac where excluded as repeated and empty/unvoiced, respectively. This dataset was randomly split into two new datasets:
  - **Dev1** (1829 files): First half.
  - **Dev2** (1830 files): Second half.

The Fisher English dataset was also used indirectly, since the BUT bottleneck extractor software was trained on it.

3. SYSTEM ARCHITECTURE

3.1. Feature extraction

Audio files where first converted to 8KHz linear PCM and then bottleneck feature vectors where extracted using the BUT bottleneck extractor software [1]. The used pre-trained NN was the so called FisherEnglish_F Bank_HL500_SBN80triphones2423, a NN trained on Fisher English with 2423 senones as targets. For speech activity detection, the bottleneck extractor’s internal energy based VAD was used.

3.2. I-vector Extraction

The Sidekit Toolkit [3] was used to create an ivector extractor. A gender independent 1024-mixture diagonal UBM was estimated by Maximum Likelihood, using the **TrainBalanced** dataset. The total variability matrix of rank 400 was estimated by 10 iterations of EM-MD on the same dataset.

3.3. Classifier

A simple generative multi-class Gaussian classifier was used to model the target languages. The distribution of language ivectors was modeled by a multivariate normal distribution \( N(\mu_l, \Sigma) \) for each target language \( l \in L \), where the full covariance matrix \( \Sigma \) was shared across all target languages. Maximum Likelihood estimates of the language dependent means \( \mu_l \) and the covariance matrix \( \Sigma \) were computed on the **Train** dataset. For each target language \( l \), the scores of an i-vector \( x \) are given by:

\[
\text{score}(x, l) = \log(N(x; \mu_l, \Sigma))
\]
Table 1. EHU fixed-primary system performance, on the NIST LRE 2017 dev set.

<table>
<thead>
<tr>
<th>Metric</th>
<th>( P_t = 0.1 )</th>
<th>( P_t = 0.5 )</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>minC</td>
<td>0.4210</td>
<td>0.1708</td>
<td>0.2959</td>
</tr>
<tr>
<td>actC</td>
<td>0.4182</td>
<td>0.1790</td>
<td>0.2986</td>
</tr>
<tr>
<td>EER</td>
<td>—</td>
<td>—</td>
<td>8.577</td>
</tr>
</tbody>
</table>

Table 2. Single threaded CPU execution time (in seconds) and amount of memory used (in Megabytes) to process 1, 10 and 100 trials of 30s of speech by each processing stage of the EHU fixed-primary system.

<table>
<thead>
<tr>
<th>Stage</th>
<th>1xTrial</th>
<th>10xTrials</th>
<th>100xTrials</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sec</td>
<td>sec</td>
<td>sec</td>
</tr>
<tr>
<td></td>
<td>MB</td>
<td>MB</td>
<td>MB</td>
</tr>
<tr>
<td>audio2bn</td>
<td>29</td>
<td>116</td>
<td>242</td>
</tr>
<tr>
<td>bn2stat</td>
<td>12</td>
<td>146</td>
<td>71</td>
</tr>
<tr>
<td>stat2ivect</td>
<td>22</td>
<td>1111</td>
<td>169</td>
</tr>
<tr>
<td>ivect2score</td>
<td>4</td>
<td>228</td>
<td>4</td>
</tr>
</tbody>
</table>

3.4. Backend

A discriminative Gaussian pre-calibration/backend was applied to the scores. The means and the common covariance matrix where initialized with their ML estimates and then further re-estimated in order to maximize the Maximum Mutual Information (MMI) criterion.

During the development phase, the Dev1 dataset was used to train the backend (Dev2 was used for validation), whereas for the final submission, the backend was trained on the full Dev dataset.

3.5. Calibration

Linear logistic regression calibration/fusion parameters were estimated on the development dataset (Dev1 during the development phase and Dev for the submission) using the FoCal Toolkit [4].

4. SYSTEM PERFORMANCE

The EHU submission consisted on a single primary system for the core fixed condition. The performance of this system on the NIST LRE 2017 dev set, using the scoring software provided by NIST is shown in Table 1.

5. PROCESSING SPEED AND MEMORY USAGE

The processing speed and memory usage was measured on a dual Xeon E5-2630v3 2.40 GHz processor, with 224 GB of RAM. Table 1 shows the processing speed and memory usage by the EHU fixed-primary system to process 1, 10 and 100 trials of 30s of speech. Comparing the speed and memory usage of 1, 10 and 100 trials is allows to detect which are the initialization requirements (i.e. the amount of time and memory required prior to process the trial). Four processing stages are reported:

- audio2bn - Bottleneck features extraction from audio file, including audio format/rate conversion and energy based VAD estimation.
- bn2stat - UBM based first and second order statistics estimation.
- stat2ivect - iVector estimation.
- ivect2score - Estimation of scores.

Note that for some stages the processing time of 1, 10 or even or 100 trials is similar, while for other stages the memory footprint does not depend on the number of trials.

6. REFERENCES


