Creating a European English Broadcast News Transcription
Corpus and System

Gerhard Backfried, Robert Hecht, Sabine Loots, Norbert Pfannerer,
Jürgen Riedler, Christian Schiefer

Speech, Artificial Intelligence, and Language Laboratories
Vienna, Austria

{Gerhard.Backfried, Robert.Hecht, Sabine.Loots}@sail-labs.at
{Norbert.Pfannerer, Juergen.Riedler, Christian.Schiefer}@sail-labs.at

Abstract
Based on BBN’s Rough’n’Ready suite of technologies used in the DARPA Hub-4 evaluations we describe the Sail-Labs Media Indexer system aiming at processing European English television broadcasts. We discuss the development of a European English broadcast news corpus, suitable for measuring performance of system components, such as speaker identification and speech recognition. We further report evaluation results on our multi-purpose test set, and outline the integration of real-time indexing into a spoken document retrieval system.

1. Introduction
The Sail-Labs Media Indexer (SMI) is aimed at indexing television news broadcasts from a variety of European English TV stations. As it turned out in initial experiments, a state-of-the-art large vocabulary continuous speech recognizer built for US English [1, 2] performs insufficiently on UK English broadcast news.

Whereas prior systems like THISL [3] focussed on BBC, we tried to cover several European English TV stations. We picked BBC, SKY, CNBC and CNN-Europe to gather a great diversity of regional dialects and non-native accents. About 85 hours of broadcast news data were collected, manually transcribed, and annotated for training and testing of our speech recognition systems.

In the next sections we describe the building steps of the various SMI-subsystems for European English, named UX English in the following. We conclude with a status report of ongoing work as well as an outlook for further system enhancements.

2. Data Collection
At present no standard corpus for European English broadcast news is available. We therefore started to collect data covering a reasonably wide range of speakers, topics, channel and background conditions. Table 1 lists the amount of collected data, of which a randomly selected part was dedicated for UX training and the complementing set for test purposes.

For every television channel selected, we chose a set of news episodes to be recorded. Among others, these included CNBC Market Watch, SKY World News, BBC World Business Report, and CNN Moneyline.

Transcription as well as annotation of the collected audio was done by means of the Transcriber tool [4], which generates a textual version of the episode in XML-format, and tags speaker turns, speaker names, genders, topics, as well as non-speech utterances.

3. Approach and Technologies
Technologies developed by BBN [1, 2] have been integrated in the SMI, and together they produce comprehensively indexed text files from the media stream input. The process is as follows, cf. Figure 5. The indexer accepts media stream through special Media Feeders. Then Speaker Change Detection (SCD) [5] marks the boundaries between different speakers’ utterances, and Automatic Speech Recognition (ASR) [6] converts the audio input into plain text. Speaker Clustering together with Speaker Classification constitute Speaker Identification (SID) [7], which identifies a set of predefined target speakers by voice. If positive identification is not possible, at least the speaker’s gender is determined. Utterances are clustered such that all utterances by one speaker are tagged with the same label throughout the episode. Named Entity Detection (NED) [8] tags entities, such as people, organizations, and locations. Finally episodes are segmented into smaller stories according to their subject and tagged with topics by Topic Detection (TD) [9]. Further details about the technologies involved are outlined below.

3.1. Speaker Change Detection (SCD)
SCD is based on phone-level decoding, the primary task of which is to label input audio as speech or non-speech. This information is used to break the audio stream into segments, each from a single speaker. It provides the speech recognizer with homogeneous input in order to allow a variety of speaker normalization techniques.

We trained several phone-class models based on US and/or UX data. Since the word error rate (WER) differences for the two data types were insignificant, we decided to use the phone models from US data solely.

<table>
<thead>
<tr>
<th>Channel</th>
<th>BBC</th>
<th>SKY</th>
<th>CNBC</th>
<th>CNN-Europe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>41.5</td>
<td>13.5</td>
<td>7.0</td>
<td>22.0</td>
</tr>
<tr>
<td>Training</td>
<td>40.0</td>
<td>5.0</td>
<td>7.0</td>
<td>19.0</td>
</tr>
<tr>
<td>Testing</td>
<td>1.5</td>
<td>2.0</td>
<td>–</td>
<td>3.0</td>
</tr>
</tbody>
</table>
3.2. Automatic Speech Recognition (ASR)

The speaker-independent, large-vocabulary speech recognizer Byblos by BBN is based on continuous density Hidden Markov Models (HMM). Training of the acoustic part includes waveform analysis, feature extraction, and parameter estimation for various types of HMM-based acoustic models. We employed a total of 106.5 hours of audio, i.e. our collected UX data (71 hours) plus the Hub-4 e97 and e98 data (35.5 hours), to train the acoustic models. In order to include additional US English data in the training, their corresponding transcriptions had to be transformed from US to British spellings. To this end we created a list of candidate words by applying a set of rules on the transcription corpus. These candidates were then sorted manually to yield the final mapping. In this way we were able to transform all words (without looking up context) to British spellings, e.g. specialize is mapped to specialise, but programme and program stay unchanged (as it turned out mapping of spellings accounts for 0.2% in WER). Furthermore, all words were checked against a large dictionary, and the remaining typos and mis-spellings were collected to form another mapping list, which in turn was applied to the transcripts.

The language model (LM) is based on word unigram, bigram, and trigram probabilities, and was trained from the Hub-4 text corpus (94–98) as well as from our TV transcripts. These two sources together contained approximately 550 million words. The word pronunciation dictionary includes all words from the transcripts plus the most frequent words from the LM corpus. The final recognition vocabulary had a size of 64k and is based on a set of 49 phones. The lexical coverage on the UX test set was 99.3%.

To evaluate ASR we compared our results with a US system, having acoustic models, that were built using the HUB-4 data set (about 210 hours of North-American broadcast news). Figure 1 depicts WERs for the UX and the US system on UX test data for individual broadcast news channels. Indeed our newly constructed UX ASR shows lower error rates: 29.8% on the average compared to 44.9% for the US system. Figure 2 displays word error rates on a US test set, composed of the e97 and e98 DARPA data. As expected, the baseline US ASR outperforms UX ASR, with 19.9% versus 28.9%. We conclude that the UX system adapts quite well to diverse channel characteristics, and will do even better when more balanced data for acoustic model training is available. In a further experiment we trained the acoustic models using 32 hours of UX data and the complete US Hub-4 data. This system retains the US baseline on the US test set and shows a WER of 35.6% on our UX set.

Figures 3 and 4 show speaker-specific WERs for speakers in the UX test set. Rather large ranges in WERs reflect the frequent and unpredictable changes from one speaker to the next, in speaking style, channel, and background conditions. For example from the first 10 speakers it is clearly visible how different both systems are able to cope with these problems. Respective average WERs are given in Table 2.

Table 2: WER per speaker type of the US and the UX system on the UX test set.

<table>
<thead>
<tr>
<th>speaker</th>
<th>target</th>
<th>non-target</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>US WER</td>
<td>39.6%</td>
<td>46.8%</td>
<td>44.9%</td>
</tr>
<tr>
<td>UX WER</td>
<td>22.2%</td>
<td>33.0%</td>
<td>29.8%</td>
</tr>
</tbody>
</table>

Target speakers were chosen from a predefined list of important speakers; see next section for details.

3.3. Speaker Identification (SID)

The SID system generates speaker identity labels for every speaker turn recognized by Speaker Change Detection. In the first step, utterances of one speaker are clustered into the same group, then these clusters are text-independently analyzed to find speakers known to the system.

For SID, we gathered speaker-time statistics of all speakers appearing on all recorded TV episodes. In addition, we retrieved a list of anchor speakers from the channels’ web-pages as well as from captions of the recorded broadcasts to obtain a set of 49 target speakers for which Gaussian mixture models were trained. Acoustic data of the remaining 211 speakers

Table 3: Precision, Recall, and the F-measure for SID on the UX test set.

<table>
<thead>
<tr>
<th></th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.982</td>
<td>0.989</td>
<td>0.985</td>
</tr>
</tbody>
</table>

Figure 1: WER of the UX versus the US ASR on the UX test set (with British spellings for the UX system and US spellings for US).

Figure 2: WER of the UX versus the US ASR on a US test set (with respective spellings).
3.4. Named Entity Detection (NED) and Topic Detection (TD)

The subsystems for NED and TD were retrained from the US corpora mapped to British spellings. A few additional names (e.g. British companies) were incorporated into the training set. After these steps, NED and TD performed just as the US systems on their respective test sets.

4. System in operation

The system described above has evolved into two versions with equal WERs. The offline indexer operates on recorded audio files and sequentially performs all the actions required to transform the input waveform into a content-based index; see Figure 5. All the measurements given in the previous section were carried out on this kind of system. For almost a year now we have been indexing approximately five hours of BBC, CNBC, CNN, SKY news shows on a daily basis, with increasing coverage over the last six months. The output of the indexer can be used to populate a database, which may then be accessed for information retrieval.

In order to accomplish the goal of real-time indexing, the architecture of the system was changed such that SCD, ASR, and SID are now performed in a pipeline. This allows the use of the same acoustic models originally built for the offline indexer. Using an 800 MHz dual-Pentium-III machine, we achieved real-time decoding with a typical memory usage of 1.5 GB. A special media-feeder starts feeding audio data into the pipeline as soon as the program to be transcribed is on the air. The resulting XML-transcriptions appear after a short initial delay. Efficient memory management allows long-term online indexing without degradation in performance, as was demonstrated on several 72-hour test runs. Exploiting its design, the pipeline indexer al-
ready was successfully integrated into a prototype multi-media archiving system [9].

5. Conclusion

Especially in the domain of speech recognition we have shown that it is possible to build a combined system, incorporating a large variety of English dialects. The UX system uses a smaller amount of US training data, and also a smaller total amount of training data. Therefore it is not unexpected that the US system outperforms the UX system on the US test set. Further experiments with varying proportions of training data and different numbers of prototypes need to be done to assess the relative effect of these influences.

6. References