# DEVELOPING HIGH PERFORMANCE ASR IN THE IBM MULTILINGUAL SPEECH-TO-SPEECH TRANSLATION SYSTEM

*Xiaodong Cui, Liang Gu, Bing Xiang, Wei Zhang and Yuqing Gao*

IBM T. J. Watson Research Center, 1101 Kitchawan Road, Yorktown Heights, NY, 10598, USA
Emails: {cuix, lianggu, bxiang, zhangwe, yuqing}@us.ibm.com

## ABSTRACT

This paper presents our recent development of the real-time speech recognition component in the IBM English/Iraqi Arabic speech-to-speech translation system for the DARPA Transtac project. We describe the details of the acoustic and language modeling that lead to high recognition accuracy and noise robustness and give the performance of the system on the evaluation sets of spontaneous conversational speech. We also introduce the streaming decoding structure and several speedup techniques that achieves best recognition accuracy at about $0.3\times$ RT recognition speed.

***Index Terms***— large vocabulary spontaneous speech recognition, multilingual speech translation, discriminative training, noise robustness, streaming mode decoding.

## 1. INTRODUCTION

IBM multilingual automatic speech-to-speech translator (MASTOR) [1][2][3] is a real-time spontaneous speech translation system that helps to remove the communication barriers between speakers who do not share a common language. As a complicated system, it integrates components such as automatic speech recognition (ASR), machine translation (MT) and speech synthesis to carry out two-way conversation. In recent years, the MASTOR system has participated in the DARPA Transtac project whose goal is to develop English/Iraqi Arabic translation systems that enable free-form communications in tactically relevant environments. The automatic speech recognition component as the crucial part of the MASTOR system is required to provide accurate, robust and low latency performance to meet the deployment demand.

In this paper, we describe the details of the acoustic and language modeling in both English and Iraqi Arabic ASR. We also address the modeling treatment to deal with background military noise. In order to deliver highly accurate recognition results in real time for translation and synthesis components, we introduce a streaming decoding structure in the Viterbi decoder and other schemes to speedup the recognition process.

The remainder of the paper is organized as follows. In Sections 2 and 3, we describe the acoustic modeling and language modeling respectively. In Section 4, the streaming decoding structure and techniques for speedup are provided. The experimental results on spontaneous speech evaluation sets are presented in Section 5 and summary is given in Section 6.

## 2. ACOUSTIC MODELING

The feature space of the acoustic model in MASTOR is created by first splicing 9 frames of 24 dimensional Mel-frequency cepstrum coefficients (MFCC) including energy and then projecting down to a 40 dimensional space by a combination of linear discriminant analysis (LDA) and maximum likelihood linear transformation (MLLT). Utterance-based cepstral mean normalization is applied.

The baseline maximum likelihood (ML) acoustic models have Gaussian mixture distribution with diagonal covariance for quinphones which are tied by a decision tree. The context-dependent acoustic models are iteratively re-estimated for 2-3 times where the LDA and MLLT are re-computed and the decision tree is rebuilt based on refined alignments produced by the previous context-dependent model.

English ASR employs gender dependent acoustic models where the male model is trained on 280 hours speech data and female model on 130 hours speech data. The male model has 4.5K quinphone states and 55K Gaussians while the female model has 4.0K quinphone states and 50K Gaussians. Iraqi Arabic acoustic model is gender independent which has 10K quinphone states and 100K Gaussians trained on 600 hours Iraqi Arabic speech data. The English acoustic models use 54 phonemes and the Iraqi Arabic acoustic model uses 33 graphemes. Both the phoneme and grapheme speech units have 3 HMM states each.

### 2.1. Discriminative Training

On top of the ML baseline acoustic model, two discriminative training strategies are applied to improve performance, namely MPE training in the back-end model domain and fMPE training in the front-end feature domain.

MPE [4] is a discriminative training criterion that has achieved superior performance over the traditional ML training. The objective function is as shown in Eq.1

$$\mathcal{F}_{MPE}(\lambda) = \sum_{r}^{R} \frac{\sum_s p_\lambda(\mathcal{O}_r|s)^\kappa p(s)^\kappa A(s,s_r)}{\sum_s p_\lambda(\mathcal{O}_r|s)^\kappa p(s)^\kappa} \tag{1}$$

where $\lambda$ are the HMM parameters, $\mathcal{O}_r$ the feature sequence of the $r$th utterance, $\kappa$ a probability scale and $p(s)$ the pre-scaled language model probability. It is an average of the "raw phone accuracy" in $A(s,s_r)$ of all possible sentences s, weighted by the sentence posterior probability. MPE criterion is in spirit the same as other discriminative objectives but the "raw phone accuracy" proves to be more effective so far performance-wise.

fMPE [5] has been shown to yield significant improvements by discriminatively training features. The mathematical treatment of this approach can be described by Eq.2.

$$y_t = x_t + M \cdot h_t \tag{2}$$

where the original feature $x_t$ is transformed into a very high dimensional space in $h_t$. This is accomplished by computing likelihood of

$x_t$ against a large number of Gaussians obtained from the acoustic model and expanding it according to its left and right acoustic context. The vector $h_t$ is then project down to the normal feature space by the project matrix $M$ and added to the original feature $x_t$ to produce the new feature $y_t$. The project matrix $M$ is trained based on the MPE criterion in Eq.1.

Working together, fMPE and MPE can achieve about 20% to 30% relative improvement over the baseline ML acoustic model, which will be shown in Section 5. For the acoustic models addressed in this paper, 4 fMPE training iterations followed by another 4 MPE training iterations are applied on the basis of the baseline ML model.

### 2.2. Multi-Style Training

Noise robustness is a persistent goal of the DARPA Transtac project due to its tactical deployment feature. The translation system is required to deliver decent performance under typical military conditions. Therefore, the ASR component has to be robust to the environment. To that end, multi-style training (MST) is adopted in acoustic modeling . In this case, 15dB noisy data are generated by adding humvee, tank and babble noise to the clean data. These three types of noise are chosen to match the military deployment environments in the Transtac project. The MST acoustic models are trained on the clean and 15dB noisy data. Accordingly, both English and Iraqi Arabic MST models have more Gaussians than the previous clean acoustic models. In English, the male and female MST models have 8K quinphone states and 90K Gaussians. In Iraqi Arabic, the MST model has 10K quinphone states and 150K Gaussians. The training procedure of the MST acoustic model is the same as previous clean model - 4 fMPE training iterations followed by 4 MPE training iterations on the basis of the ML acoustic model.

It was observed from experiments and Transtac evaluations that the speech signals recorded via a high quality microphone can be actually very clean (close to 18-20 dB) even in a relatively noisy environments (e.g. field evaluation condition). Therefore, MST acoustic models will only be invoked in MASTOR when the SNR of the recorded input signals is below certain threshold, which can be accomplished by environment detection.

### 2.3. Feature Adaptation

During recognition, online adaptation based on feature space maximum likelihood linear regression (fMLLR) [6] is used to adapt the system to the acoustic features of the speaker. The adaptation is a linear transformation in the MFCC feature space which has the form in Eq. 3.

$$\hat{x}_t = A \cdot x_t + b \tag{3}$$

where the transformation matrix $A$ and bias $b$ are estimated online incrementally during the course the speaker uses the system. The process is unsupervised and the statistics collected from all the previous utterances contribute to the estimation of the transformation of the current utterance.

Compared to the maximum likelihood linear regression (MLLR) [7] on adapting back-end acoustic model parameters, fMLLR offers a computational advantage for rapid yet effective adaptation. From the performance perspective, fMLLR yields about 10% additional relative improvement on top of fMPE and MPE trained acoustic model.

## 3. LANGUAGE MODELING

The Viterbi decoder in MASTOR operates on a static graph that is based on finite state automata (FSA) [8]. The static graph incorporates acoustic decision tree, language model (LM) and dictionary where the last two items significantly affect the size of the graph and consequently the memory consumption during run-time. Therefore, tradeoff has to be made among accuracy, memory and speed when dealing with language modeling issues.

Both English and Iraqi Arabic ASRs use trigram in MASTOR. The English ASR has a vocabulary of 32K words. The LMs are generated by interpolating an in-domain LM trained on text corpus with less than 15M words with a general domain LM that is trained on a much larger text corpus. Both word-based and class-based LMs have been investigated with several n-gram pruning strategies. Since MASTOR can operate on multiple ASR engines as will be discussed in Section 4, various LMs are adopted in MASTOR on the multiple engines to cope with different situations. The final LMs have about 4M to 7M n-grams. The LMs in Iraqi Arabic ASR are trained on in-domain data and have around 6M n-grams with different vocabulary sizes (from 130K to 180K words) after pruning.

## 4. STREAMING DECODING STRUCTURE AND SPEEDUP

### 4.1. Streaming Decoding Structure

Fig.1 demonstrates the signal processing flow of MASTOR and particularly its ASR decoding infrastructure. There are usually two types of ASR decoding structure: utterance mode and streaming mode. In utterance mode, the input speech is recognized only when the whole utterance is recorded and sent to the ASR engine. Therefore, the context information between various speech segments may be exploited either implicitly or explicitly to achieve highest recognition accuracy. However, this will surely lengthen the system response time which is simply not acceptable in speech-to-speech translation applications because the high ASR delay will greatly reduce the message transfer rate of the multilingual conversation and even break the conversation into pieces. In streaming mode, the input speech is segmented into small (such as 250ms) chunks and sent to the ASR engine chunk by chunk. The ASR decoder generates the best recognition output based on both the current speech chunk and the previous recognition hypothesis stored in a Viterbi lattice. By reducing the size of each chunk, the approach can achieve speech recognition at very low latency. However, because of the lack of forward-looking in both front-end analysis and lattice search, the resulting recognition accuracy is often sub-optimal. To achieve highly accurate speech recognition with low latency, we propose a new way of decoding in streaming mode. Different from the traditional way of streaming mode recognition, we break the recognition process into multiple layers. In particular, we define six layers in the MASTOR system, i.e., MFCC, LDA, Cepstral Mean Normalization, fMLLR, fMPE and Viterbi search. The streaming mode is applied to each layer with a layer-dependent chunk size. If the chunk size equals the utterance length, then the decoding in that layer is equivalent to the utterance mode decoding. The smaller the chunk size in a layer, the more likely its decoding performs in a streaming mode. The size of chunks for each layer is optimized on a dev set to minimize both recognition accuracy loss compared to the utterance mode decoding and the corresponding recognition latency.

### 4.2. Decoding Speedup

The proposed streaming mode ASR decoder is further optimized for decoding speed. The computational overhead is minimized among multiple decoding layers and multiple speech chunks. A new algorithm called maximum probability improvement estimation (MPIE) is proposed in fMPE computation that estimates posteriors of each Gaussian distribution between current speech frame and previous frames in order to reduce explicit computations of posteriors for a large number of Gaussians. A cluster-based fast Gaussian mixture computation scheme is applied to speedup likelihood computation during Viterbi search in the lattice. Moreover, all the six layers of streaming mode decoding are programed to support parallel computing, which may lead to up to 100% speedup on an Intel Duo-Core based computer.

### 4.3. Multiple Engines

The MASTOR system is designed for high-performance speech-to-speech translation even in very severe situations. Therefore, the speech recognition accuracy should be robust to user gender, user accents, noise backgrounds, and speech translation domains. On the other hand, both acoustic models and language models work best if they are trained for a matched group of people in a matched noise background and within a matched speech translation domain. Not surprisingly, however, these models usually perform worst if they are used in non-matched conditions. It is hence an open challenge to design highly accurate and also highly robust ASR engines. We proposed and implemented a novel way to attack the above challenge in the MASTOR system. Instead of fulfilling ASR task by a single engine, we applied multiple ASR engines at the same time. Each engine has its own acoustic model and language model. Some of these models can be shared between various ASR engines to optimize memory usage. During speech recognition, the input speech is sent to the multiple engines simultaneously. The speech utterance is then recognized by these ASR engines in parallel. Each engine then returns its own best recognition hypothesis. The ASR hypothesis from these multiple engines are further unified, sorted and displayed as N-best ASR hypothesis to the users. A ROVER algorithm is designed and implemented to generate the best ASR result based on these N-best ASR hypothesis. Note that the approach of multiple ASR engines will inevitably increase both computational complexity and memory consumption. Since the Viterbi decoder applied in our MASTOR system involves both high computational complexity and memory consumption, the resulting ASR component may easily pass the limit of available system memory and acceptable ASR responding time, even if only 2 ASR engines are adopted. Alternatively, we combine our best Viterbi decoder with multiple low-footprint IBM ViaVoice engine using stack decoding , whose computational cost and memory requirement are extremely low. This solution achieved satisfactory results in both offline ASR experiments and DARPA real-time evaluations.

## 5. PERFORMANCE

In this section, we will present the experimental results in terms of recognition accuracy and speed. There are a number of test sets that the ASR component is evaluated.

### 5.1. Recognition Accuracy

Table 1 shows the word error rate (WER) of English ASR with the male (55K Gaussians) and female (50K Gaussians) acoustic models

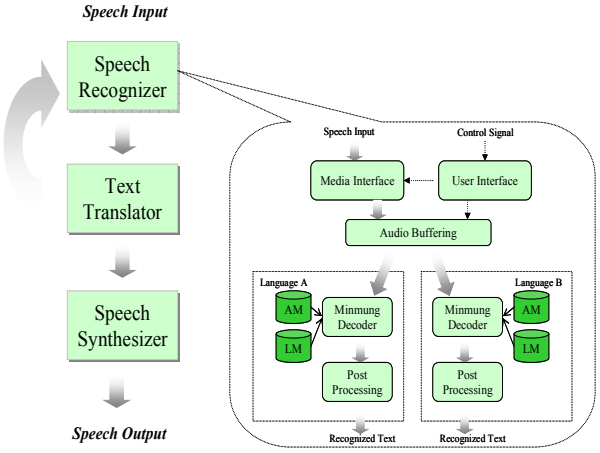


**Fig. 1**. Diagram of the signal processing flow of MASTOR with a detailed description on the decoding infrastructure of ASR.

on three test sets that are collected in the DARPA Transtac project. The three test sets all contain spontaneous speech in which "lab" and "field" have Transtac January 2007 evaluation data recorded during conversation in laboratory and field conditions respectively and "offline" is composed of Transtac January 2007 evaluation data recorded offline. The majority of the speakers in the three test sets are male speakers. (There are only 5 female speakers in total who are all in the "offline" test set. They are not evaluated in the experiments.) Each set has about 350 utterances. The language model used in the experiments is word-based interpolated LM discussed in Section 3. English female is evaluated on another test set "spont" which has 205 spontaneous utterances in total. Table 2 shows the performance on Iraqi Arabic acoustic model (100K Gaussians). The "lab", "field" and "offline" test sets are the Iraqi Arabic counterpart of those in English in Table 1.

| Test Condition | male | | | female |
|---|---|---|---|---|
| | lab | field | offline | spont |
| ML | 11.9 | 9.9 | 23.0 | 17.3 |
| fMPE+MPE | 7.9 | 6.8 | 19.3 | 14.7 |
| fMPE+MPE+fMLLR | 6.8 | 5.9 | 18.2 | 13.9 |

**Table 1**. Word error rate (WER) of English acoustic models on spontaneous speech test sets.

| Test Condition | lab | field | offline |
|---|---|---|---|
| ML | 29.1 | 25.1 | 31.9 |
| fMPE+MPE | 24.8 | 21.6 | 28.8 |
| fMPE+MPE+fMLLR | 22.5 | 19.7 | 27.8 |

**Table 2**. Word error rate (WER) of Iraqi Arabic acoustic model on spontaneous speech test sets.

From the tables, it can be observed that fMPE and MPE together obtain about 20%-30% relative word error rate reduction compared to the baseline ML model. Furthermore, fMLLR yields additional relative 10% improvement on top of fMPE and MPE. This has been the best reported performance in recognition accuracy for the Transtac 2007 January evaluation.

Tables 3 and 4 give the performance of different LMs in English

and Iraqi Arabic. Since the ASR component of MASTOR employs a Viterbi decoder using a static graph which integrates dictionary, LM and acoustic decision tree, the size of LM is an important factor as it will greatly affect the size of the static graph. In Table 3, English word-based and class-based interpolated LMs are compared. The acoustic models are trained by fMPE and MPE and evaluated witout fMLLR. LM2 is pruned more heavily than LM1. It can be observed that word-based and class-based LMs obtain comparable results with class-based LM being slightly better for most cases. Table 4 are results for Iraqi Arabic LMs which are trained with in-domain data. The numbers in the parentheses are the vocabulary size after pruning with different thresholds. The final LM is chosen for Transtac evaluation is built on 130K vocabulary based on the tradeoff between performance and LM size.

| Test Condition | male | | | female |
|---|---|---|---|---|
| | lab | field | offline | spont |
| word-based LM1 | 7.5 | 6.6 | 17.8 | 14.2 |
| word-based LM2 | 7.5 | 6.8 | 18.4 | 14.4 |
| class-based LM1 | 7.3 | 6.7 | 18.2 | 14.1 |
| class-based LM2 | 7.4 | 6.5 | 18.4 | 14.1 |

**Table 3**. Word error rate (WER) of English language models on spontaneous speech test sets.

| Test Condition | lab | field | offline |
|---|---|---|---|
| LM(180K) | 22.0 | 19.8 | 21.6 |
| LM(160K) | 22.3 | 19.5 | 22.0 |
| LM(130K) | 22.3 | 19.4 | 22.0 |

**Table 4**. Word error rate (WER) of Iraqi Arabic language models on spontaneous speech test sets.

The high ASR accuracy delivered by the recognition system also helped to achieve the best end-to-end system performance of IBM English-Iraqi Arabic translation system in Transtac July 2007 evaluation. IBM system has achieved the highest scores in both high level concept transfer rate and low level concept transfer rate in the evaluation.

### 5.2. Recognition Speed

Fig.2 illustrates the trend of improvement in recognition speed in terms of real time factor as speedup schemes adding in. The experiments are conducted on the English male "lab" test set using the 55K Gaussian male acoustic model and word-based interpolated language model. The real time factors are measured on IBM T60p laptop with 2.33G Intel Duo-Core CPU. With carefully tuned decoding parameters, both English and Iraqi Arabic can run at about the same speed.

The speedup techniques as shown in the figure include parallel computation (double threading), fast Gaussian computation, fast search and fast feature extraction. Each of the technique contributes about 20% relative improvement in speed and the trend is consistent for both English and Iraqi Arabic. Overall, the speed is reduced from original 1.4×RT to the final 0.3×RT.

## 6. SUMMARY

In this paper, we describe the details of the acoustic and language modeling in both English and Iraqi Arabic ASR. We also address the modeling treatment to deal with background military noise. In order to deliver recognition results in real time for translation and synthesis components, we introduce a streaming decoding structure in the Viterbi decoder and schemes to speedup the recognition process which include parallel computation, fast Gaussian computation, fast search and fast feature extraction.

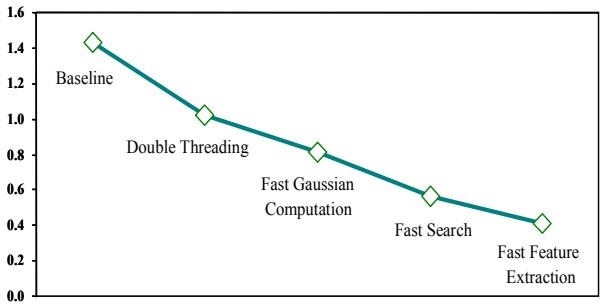


**Fig. 2**. Improvement of recognition speed in real time factor through speedup techniques.

## 7. ACKNOWLEDGEMENTS

This material is based upon work supported by the DARPA Transtac project. We would like to thank H. Soltau and D. Povey for helpful discussions on acoustic model training.

## 8. REFERENCES

[1] Y. Gao, B. Zhou, L. Gu, R. Sarikaya, H.-K. Kuo, A.-V.I. Rosti, M. Afify, and W. Zhu, "IBM MASTOR: Multilingual automatic speech-to-speech translator," *Proc. of ICASSP*, pp. 1205–1208, 2006.

[2] B. Zhou, D. Dechelotte, and Y. Gao, "Two-way speech-to-speech translation on handheld devices," *Proc. of ICSLP*, pp. 1637–1640, 2004.

[3] L. Gu, Y. Gao, F. Liu, and M. Picheny, "Concept-based speech-to-speech translation using maximum entropy models for statistical natural concept generation," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 2, pp. 377–392, 2006.

[4] D. Povey and P.C. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," *Proc. of ICASSP*, pp. 105–108, 2002.

[5] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: Discriminatively trained features for speech recognition," *Proc. of ICASSP*, pp. 961–964, 2005.

[6] Y. Li, H. Erdogan, Y. Gao, and E. Marcheret, "Incremental online feature space mllr adaptation for telephony speech recognition," *Proc. of ICSLP*, pp. 1417–1420, 2002.

[7] C. J. Legetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech and Language*, vol. 9, no. 2, pp. 171–186, 1995.

[8] G. Saon, D. Povey, and G. Zweig, "Anatomy of an extremely fast lvcsr decoder," *Proc. of Interspeech*, pp. 549–552, 2005.