IBM MASTOR: MULTILINGUAL AUTOMATIC SPEECH-TO-SPEECH TRANSLATOR

Yuqing Gao, Bowen Zhou, Liang Gu, Ruhi Sarikaya, Hong-kwang Kuo,
A-V.I. Rosti, Mohamed Afify, Weizhong Zhu

IBM T.J. Watson Research Center, Yorktown Heights, NY 10598

ABSTRACT

In this paper, we describe the IBM MASTOR systems which handle spontaneous free-form speech-to-speech translation on both laptop and hand-held PDAs. Challenges include speech recognition and machine translation in adverse environments, lack of data and linguistic resources for under-studied languages, and the need to rapidly develop capabilities for new languages. Importantly, the code and models must fit within the limited memory and computational resources of hand-held devices. We describe our approaches, experience, and success in building working free-form S2S systems that can handle two language pairs (including a low-resource language).

1. INTRODUCTION

Automatic speech-to-speech (S2S) translation has a lot of potential practical applications for both laptops and hand-held devices, promising to break down communication barriers between people who do not share a common language and enable instant oral cross-lingual communication. At the same time, the development of accurate and efficient S2S translation systems poses a lot of challenges. This is especially true for colloquial speech and resource deficient languages.

Among the challenges for speech recognition and translation for under-studied languages, there are two main issues: 1) Lack of appropriate amount of speech data that represent the domain of interest and the oral language spoken by the target speakers, resulting in difficulties in accurate estimation of statistical models for speech recognition and translation. 2) Lack of linguistic knowledge realization in spelling standards, transcriptions, lexicons and dictionaries, or annotated corpora. Therefore, various different approaches have to be explored.

Another critical challenge is to embed complicated algorithms and programs into small devices for mobile users. A hand-held computing device may have a CPU of 256MHz and 64MB memory; to fit the programs, as well as the models and data files into this memory and to operate the system in real-time are tremendous challenges [1].

S2S translation efforts at IBM have focused on the challenges of bidirectional free-form natural communication between the participating parties. Initial work led to full bidirectional S2S systems for English and Mandarin Chinese, and the emergence of the MASTOR multilingual S2S development environment [2]. More recently, we have significantly broadened our experience and efforts to very rapidly develop systems for under-studied languages, such as regional dialects of Arabic.

In this paper, we will describe our approaches for developing main components, such as speech recognition and translation, and the design of the overall system for different computing platforms [1]. Various statistical approaches (e.g., LM approaches [3], [4], translation approaches, [5] for concept-based, [6] for FST-based) are explored and used to solve different technical challenges. We will compare our experience working with a relatively resource-rich language like Mandarin Chinese with a language like colloquial Arabic dialect that does not have any existing resources. We will show how we addressed the challenges that arise when building automatic speech recognition (ASR) and machine translation (MT) for colloquial Arabic, give a brief system description, highlight the difficulties that we faced in migrating the original system to the hand-held computer, and report our results.

2. MAJOR TECHNOLOGY COMPONENT - ASR

2.1. Acoustic Models

Acoustic models and the pronunciation dictionary greatly influence the ASR performance. In particular, creating an accurate pronunciation dictionary poses a major challenge when a new language is involved. Deriving pronunciations for resource-rich languages like English or Mandarin is relatively straightforward using existing dictionaries or letter to sound models. In certain languages such as Arabic and Hebrew, the written form does not typically contain short vowels which a native speaker can infer from context. Deriving automatic phonetic transcription for speech corpora is thus difficult. This problem is even more apparent when considering colloquial Arabic, mainly due to the large number of irregular dialectal words.

One approach to overcome the absence of short vowels is to use grapheme based acoustic models. These lead to straightforward construction of pronunciation lexicons and hence facilitates model training and decoding. However, the same grapheme may lead to different phonetic sounds depending on its context. This results in less accurate acoustic models. For this reason we experimented with two other different approaches. The first is a full phonetic approach which uses

Thanks to DARPA for funding the work.

short vowels, and the second uses context-sensitive graphemes for the letter "A" (Alif) where two different phonemes are used for "A" depending on its position in the word.

To perform vowelization, which is necessary for phonetic transcription, we used a mix of dictionary search and a statistical approach. The word is first searched in an existing vowelized dictionary, and if not found it is passed to the statistical vowelizer[7].

Acoustic modeling for both the laptop and hand-held systems is based on the IBM ViaVoice product engine. This highly robust and efficient framework uses rank based acoustic scores [8] which are derived from tree-clustered context dependent Gaussian models. These acoustic scores together with n-gram LM probabilities are incorporated into a stack based search algorithm to yield the most probable word sequence given the input speech.

2.2. Language Models

Language modeling (LM) of the probability of various word sequences is crucial for high-performance ASR of free-style open-ended conversational systems. N-gram LMs need a large amount of text data to represent the domain word usage distribution. In applications such as S2S, such a database often does not exist for the languages and domains of interest. Standard N-grams may suffer from sparse data. Significant challenges include how to estimate the word sequence probability distributions with very limited amount of training data from the specific application domains. These challenges are even stronger for under-studied languages.

Our approaches to build LMs fall into three categories: 1) obtaining additional training material automatically; 2) interpolating domain-specific LMs with other LMs; 3) improving distribution estimation robustness and accuracy with limited in-domain resources. Automatic data collection and expansion is the most straight-forward way to achieve efficient LM, especially when little in-domain data is available. For resource-rich languages such as English and Chinese, we retrieve additional data from the World Wide Web (WWW) to enhance our limited domain specific data, which shows significant improvement [4]. We also combine different LMs trained using various resources using interpolation-based methods. We also use semantic concept based LMs[3]. However for colloquial Arabic dialect, because of severe data sparseness, some of the approaches may not apply. We have worked out approaches suitable for very limited data.

We build three separate models using the small in-domain corpus for Arabic: 1) a standard trigram language model with deleted interpolation for smoothing; 2) a class-based language model after generating 13 classes from the data. Typical classes include Arabic person names, transliteration of English person names, city names, currency, color, months, dates, digits, etc. These classes are supplemented with words that do not appear in the training data; 3) a morpheme language model after performing a limited morphological analysis on the corpus. After analyzing the lexicon and text data we selected 16 prefixes, which are commonly used in the regional Arabic dialect, and re-tokenized those words that contain these prefixes into prefix and stem parts. The purpose of this analysis is to increase the language model coverage against unseen words. The prefix list is determined by checking words in the vocabulary whether they accept these prefixes to form new legitimate colloquial Arabic words. For example, (in Buckwalter transliteration) bAl$hr may not exist in the training data, but individual prefix bAl and stem $hr do exist. Therefore the morpheme LM will allow the decoding of bAl$hr (after merging prefix and stem by checking the word list). The final language model is the interpolation of these 3 models. The interpolation weights are optimized on a held-out test data.

3. MAJOR TECHNOLOGY COMPONENT - TRANSLATION

3.1. NLU/NLG-based Speech Translation

Statistical machine translation methods translate a sentence W in the source language into a sentence A in the target language by using a statistical model that estimates the probability of A given W, i.e. p(A|W). Conventionally, p(A|W) is optimized on a set of pairs of sentences that are translations of one another. To alleviate the data sparseness problem and, hence, enhance both the accuracy and robustness of estimating p(A|W), we proposed a statistical concept-based machine translation paradigm that predicts A with not only W but also the underlying concepts embedded in W and/or A. As a result, the optimal sentence A is picked by first understanding the meaning of the source sentence W.

Let C denote the concepts in the source language and S denote the concepts in the target language, our proposed statistical concept-based algorithm should select a word sequence $\hat{A}$ as

$$\begin{aligned}\hat{A} &= \arg\max p(A \mid W) \\ &= \arg\max\{\sum\sum p(C \mid W)p(S \mid C,W)p(A \mid S,C,W)\}\end{aligned}$$

where the conditional probabilities p(C|W), p(S|C,W) and p(A|S,C,W) are estimated by the Natural Language Understanding (NLU), Natural Concept Generation (NCG) and Natural Word Generation (NWG) procedures, respectively. The probability distributions can be estimated and optimized upon a pre-annotated bilingual corpus by various methods such as Maximum Likelihood, Maximum Entropy, Decision Tree, etc. Without losing generality, in our proposed work, p(C|W) is estimated by a decision-tree based statistical semantic parser, and p(S|C,W) and p(A|S,C,W) are estimated by maximizing the conditional entropy as depicted in [5] and [9], respectively. In particular, p(S|C,W) is estimated by maximizing the conditional entropy

$$H_{NCG}(p) \equiv -\sum_{(C,S,W)\in X} p(C,W)p(S|C,W)log\{p(S|C,W)\},$$

and p(A|S,C,W) is estimated by maximizing the conditional entropy

$$H_{NWG}(p) \equiv -\sum_{(A,C,S,W)\in Y} p(S,C,W)p(A|S,C,W)log\{p(A|S,C,W)\}$$

where X and Y is the training data that consists of both the word sequence and semantically-annotated treebanks in the source and the target languages. Accordingly, we have proposed a series of algorithms to improve the performance of NCG, including forward-backward modeling, multi-dimensional bilingual context-dependent feature sets and confidence-based processing.

3.2. Phrase-based WFST Speech Translation

We explored statistical translation approaches trained from unannotated parallel corpus data to overcome the need for semantic annotation and to speed up the development cycle. We recently significantly extended our previous work of constrained phrase-based approach [6] to free phrase-based translation approach within the framework of Weighted Finite State Transducer (WFST) to accommodate the following design considerations for multiple platforms including PCs and handheld PDAs. First, the footprint of the translation component needs to be sufficiently compact to be hosted on computing devices with a small amount of memory; secondly, the translation engine is required to be computationally efficient to achieve fast translation to practically aid the face-to-face speech communication; and finally, the architecture of the translation component should allow efficient interactions with the speech interface, especially the speech recognition component, for possible future joint system optimization.

In this new work, all major models, including the automatically learned phrase segmentation and phrase translation models, are represented in WFST's and composed into a unified lattice, which is able to be globally optimized using standard finite state algorithms based on some specific designs. A Viterbi decoder is developed to meet the challenges of fast decoding and small memory footprints. The details of the translation system design will be described in [10].

There are two significant advantages of our design and implementation of the translation component. First, our system achieves a very fast translation speed. On a regular Linux box with an Intel Xeon 2.40GHz processor, the translation engine obtains an average translation speed of around $7,000$ words per second for multiple tasks (see Sec. 5). When running on a handheld device such as a PDA with an Xscale 400MHZ processor, it still operates at a speed of several hundreds of words per second. Secondly, our design enables us to achieve a converged translation engine for multiple platforms. That is, the proposed architecture allows us to implement the translation engine only once and can be easily compiled for different platforms without any additional porting efforts. For example, the engines compiled from the same code base are able to run on both standard PCs (with Linux or Windows OS) and a PDA device (with WinCE OS) that has only 64 MB physical memory. This is achieved through the complete fixed-point arithmetic used in the Viterbi decoder for our WFST-based translation system, as well as a memory efficient search algorithm implementation. Details of the converged translation architecture will be described in [11].

4. SYSTEM DESCRIPTION

The general framework of our MASTOR system has components of ASR, MT and TTS. The cascaded approach of ASR, MT and TTS allows us to deploy the power of the existing advanced speech and language processing techniques, while concentrating on the unique problems in S2S translation.

Acoustic models for English and Mandarin baseline are developed for large-vocabulary continuous speech and trained on over 200 hours of speech collected from about 2000 speakers for each language. However, the Arabic dialect speech recognizer was only trained using about 50 hours of dialectal speech. The training data for Arabic consists of about 200K short utterances. Large efforts were invested in initial cleaning and normalization of the training data because of a large number of irregular dialectal words and variations in spellings. We experimented with three approaches for pronunciation and acoustic modeling: i.e. grapheme, phonetic, and context-sensitive grapheme as described in Sec. 2.1. We found that using context-sensitive pronunciation rules reduces the WER of the grapheme based acoustic model by about 3% (from 36.7% to 35.8%). Based on these results, we decided to use context-sensitive grapheme models in our system. The Arabic acoustic model contains about 30K Gaussians and 2K leaves.

The Arabic language model (LM) is an interpolated model consisting of a trigram LM, a class-based LM and a morphologically processed LM, all trained from a corpus of a few hundred thousand words. We also built a compact language model for the hand-held system, where singletons are eliminated and bigram and trigram counts are pruned with increased thresholds. The LM footprint size is 10MB.

There are two approaches for translation. The concept based approach uses natural language understanding (NLU) and natural language generation models trained on annotated corpus. In parallel, phrase-based finite state transducer is another approach which uses raw text data. For the translation between English and Chinese, both methods are used in parallel to improve the translation accuracy. For the translation between English and Arabic, only the finite state transducer approach is used due to the lack of the resources for concept annotation. The English and Chinese NLU and NL models are trained from 15k annotated sentences, while the WFST models are trained from 60k sentences raw text data. Benefiting from our converged design philosophy, the complete end-to-end speech translation system is able to be built for a handheld device (e.g., iPaq 3500). Continuing from our previous efforts in building the handheld two-way speech translation system [1], this work is extended in the following ways. First, the translation component is the converged engine (see Sec. 3.2) using phrase-based translation models. Secondly, the system is built upon on WinCE platforms rather than on embedded Linux, and the IBM ViaVoice recognition engine thus

is ported to WinCE [1] in a similar fashion to that described in [1]; In addition, the text-to-speech component is based on embedded concatenative TTS technologies [7].

5. EXPERIMENTAL RESULTS

The English-Mandarin recognition and translation experiments were done on the DARPA CAST Aug'04 offline evaluation data, which has an English script of 130 sentences and a Chinese script of 73 sentences for medical domain. Each script was read by 4 speakers. The recognition word error rate for English is 11.06%, while the character error rate for Mandarin is 13.60%, both are run on speaker-independent models. The translation experiments are done on both clean text and the ASR decoded scripts. The 4-gram Bleu score results measured using 8 human translations as references are shown in Table 1. The oracle scores show that if one can combine the translation results from these two different approaches, the accuracy can be further improved significantly. Currently we present two alternate translations to users in the real-time system to enhance the communications. It is very useful to notice that the translation results generated by our two approaches are always consistent in meaning.

	En-to-Cn		Cn-to-En	
Input	Clean	ASR	Clean	ASR
NLU/NLG	0.578	0.513	0.276	0.245
WFST	0.572	0.504	0.276	0.246
NLU/NLG+WFST (Oracle)	0.691	0.606	0.365	0.342

Table 1. BLEU scores of English-Mandarin translation

English-Arabic experiments are done on several 3-person cross-lingual conversations (a subset of DARPA development set). In each dialog, an English speaker and an Arabic speaker were talking to each other via a human interpretor. We extracted 395 English utterances and 200 colloquial Arabic utterances from the dialogs. Three human translation references are created for measuring the BLEU score purpose. The results are shown in Table 2. Since the data is spontaneous conversational speech, the recognition WERs for both English and Arabic are as high as 40%. The BLEU scores of English-to-Arabic is slightly lower than that of Arabic-to-English. One possible reason is that spelling of words in colloquial Arabic dialect is not standardized (more variations for the same word), which can lead to a low BLEU score. Another observation is that the ASR errors degrade the BLEU score more significantly for English-to-Arabic. Although the ASR WERs look similar for English and Arabic, we notice that the WER of English content words is higher than that of Arabic. A possible reason is that the English acoustic model is not trained from spontaneous speech, while the Arabic acoustic model is trained with more conversational style speech mainly from in-domain data. Another reason is that the English test data includes speech from non-native English speakers (interpreters), which led to high WERs.

	ASR WER	BLEU (Clean)	BLEU (ASR)
En-to-Ar	42.92%	0.4748	0.2506
Ar-to-En	41.68%	0.4206	0.3240

Table 2. BLEU scores of English-Arabic translation

6. CONCLUSIONS

We described the framework of the IBM MASTOR system and the various technologies used in building major components for languages with different data resource levels. The technologies have enabled the successful building of real-time S2S systems on low computation resource platforms (200 MHz CPU and 64 MB memory) for two language pairs, English-Mandarin Chinese, and English-Arabic dialect. In the latter case, we also developed approaches which lead to very rapid (in the matter of 3-4 months) development of systems using very limited language and domain resources. We are working on improving spontaneous speech recognition accuracy and more naturally integrating two translation approaches.

7. REFERENCES

[1] B. Zhou et al, "Two-way speech-to-speech translation on handheld devices," in *Proc. ICSLP'04*, 2004.

[2] Y. Gao et al, "MARS: A statistical semantic parsing and generation based Multilingual Automatic tRanslation system," *Machine Translation*, pp. 185–212, 2004.

[3] H. Erdogan et al, "Using semantic analysis to improve speech recognition performance," *Computer Speech and Language*, vol. 19, pp. 321–343, 2005.

[4] R. Sarikaya et al, "Rapid language model development using external resources for new spoken dialog domains," in *Proc. ICASSP'05*, 2005.

[5] L. Gu et al, "Forward-backward modeling in statistical natural concept generation for interlingua-based speech-to-speech translation," in *Proc. IEEE ASRU Workshop'03*, 2003.

[6] B. Zhou et al, "Constrained phrase-based translation using weighted finite-state transducers," in *Proc. ICASSP'05*, 2005.

[7] O. Emam et al, "A framework for Arabic concatenative tect-to-speech synthesis," in *ICASSP'06*, 2006, submitted.

[8] L. R. Bahl et al, "Robust methods for using context-dependent features and models in a continuous speech recognizer," in *Proc. ICASSP-94*, 1994.

[9] L. Gu et al, "Use of maximum entropy in natural word generation for statistical concept-based speech-to-speech translation," in *Proc. Interspeech'05*, 2005.

[10] B. Zhou et al, "Fast phrase-based statistical translation using FST's," in *In Preparation*, 2005.

[11] B. Zhou et al, "A unified scalable real-time statistical translation for high- and low-end computing devices," in *In Preparation*, 2005.

[1] Thanks to colleagues at IBM Japan for the contribution.