Language Identification for Speech-to-Speech Translation

Daniel Chung Yong Lim, Ian Lane

Language Technologies Institute, Carnegie Mellon University, Pittsburgh, USA
{cdlim, ianlane}@cs.cmu.edu

Abstract

This paper investigates the use of language identification (LID) in real-time speech-to-speech translation systems. We propose a framework that incorporates LID capability into a speech-to-speech translation system while minimizing the impact on the system’s real-time performance. We compared two phone-based LID approaches, namely PRLM and PPRLM, to a proposed extended approach based on Conditional Random Field classifiers. The performances of these three approaches were evaluated to identify the input language in the CMU English-Iraqi TransTAC system, and the proposed approach obtained significantly higher classification accuracies on two of the three test sets evaluated.

Index Terms: language identification, speech-to-speech translation, conditional-random-field classifiers, parallel phone recognizer

1. Introduction

Current Speech-to-Speech (S2S) translation systems are used for bi-directional, bilingual communication. When developing these systems it is assumed that there will be two participants each speaking one of the two predefined languages that the system was built to handle. The goal of S2S translation is to facilitate a conversation between them. The conversation is usually carried out in a turn-based fashion:

- First the primary user who is carrying the device speaks to the system in L_p.
- The system invokes the appropriate chain of Automatic Speech Recognition (ASR), Machine Translation (MT), and Text to Speech synthesis (TTS) modules to translate the speech from L_p into the secondary language L_s.
- Next the secondary user may respond to the output and speak to the system in L_s.
- The system will then carry out S2S translation from L_s to L_p.

Figure 1 shows a block diagram of a typical S2S translation system.

In current S2S translation systems, L_s is preset to one specific language, for example Iraqi in the CMU English-Iraqi TransTAC system [1]. However, if the system is deployed to a location where multiple languages are present, the language of the secondary user (L_s) would need to be identified before a dialog could take place. Once L_s is correctly identified, S2S translation can be performed as in current systems because L_s is unlikely to change within the course of a dialog. One approach to determine L_s automatically is to perform Language Identification (LID) on the first few utterances from the secondary user and then use this result to select the appropriate modules within the S2S translation system which will be applied in the remainder of the dialog.

There are two application scenarios where this framework is directly applicable. First, when portable translation systems, such as the CMU TransTAC system, are deployed to locations where the local population uses multiple languages. For example, extending the system to operate in Afghanistan where both the Dari and Pashtu languages are common. And second, when S2S translation systems are developed to operate in locations where there are a large number of foreign visitors from numerous countries, such as at an information desk in an airport or tourist spot. In the second case, rather than having multiple systems for each secondary language, a single system that could select the appropriate language automatically based on the input speech of the visitor would suffice.

By incorporating LID into S2S translation systems, not only will the capabilities of the system be expanded, but the system will also be more intuitive to use, as the user will not have to identify the required language at the start of each dialog.

In this paper we propose a framework for incorporating LID into the CMU TransTAC system, and evaluated the performance of LID for English-Iraqi Arabic identification, focusing on using phone-based approaches.

2. Language Identification in Speech-to-Speech Translation

One approach to incorporate LID into a S2S translation system is to make use of the automatic speech recognizers already available within the system. The user’s utterance is run through each of the speech recognizers in parallel and the input language will be determined by the recognizer that returns the highest ASR score. As reported in [2], this approach typically obtains high accuracy for LID. However, this approach is not suitable for applications that have real-time constraints, as all speech recognizers must complete decoding the input utterance before a decision can be made. When there is a mismatch in the language of the utterance and the speech recognizer, the time taken for decoding is typically significantly longer than real-time. Furthermore, this approach is not feasible when there are more than two or three languages to identify due to the high computational cost.

An alternative is to perform LID using other approaches that incur lower computational cost, such as phone-based techniques. Once LID is completed, the appropriate S2S process is then invoked. The drawback is that the S2S process can only begin after LID has completed. This causes a delay in the response of the system.

In turn we propose a framework for incorporating LID into a S2S translation system with the objective of minimizing any delay that the usage of LID might introduce. We make two assumptions, the first being that the language pair used rarely changes in a conversation. The second assumption is that the

![Figure 1: Block diagram of a S2S translation system.](image-url)
participants usually take equal turns to speak to one another, i.e. it is seldom the case that one user will conduct a long monologue before letting the other user speak.

In the proposed framework, at the beginning of each turn, we first predict \( L_T \), the language that will be spoken in that turn, by observing the previous conversation patterns. As a user begins to speak, we run his utterance through the LID module and the speech recognizer for \( L_T \) simultaneously. When the LID module completes running, we compare its result \( \text{LID} \), with \( L_T \). If the two are the same, i.e. the prediction is correct, we continue with the rest of the S2S processes to translate \( L_T \). However, if the prediction is wrong, we terminate the speech recognition of \( L_T \) if it is still running, and start a new S2S process for translating \( \text{LID} \). Figure 2 shows the flow of events for an example where a switch in the S2S processes is necessary.

3. Language Identification Approaches

Phone-based approaches [3] are a class of technique for LID that incurs lower computational cost than using full ASR. Phone-based approaches are motivated by the observation that some sequences of phones that exist in one language rarely exist in another language. Thus the frequency of occurrence of phone sequences in each language can be modeled and used for identification. We evaluated the basic Phone Recognition followed by Language Modeling (PRLM) and Parallel PRLM (PPRLM) approaches described in [3] for the identification of English and Iraqi Arabic. We also propose an extended approach where individual phone recognition scores are integrated into a Conditional Random Fields (CRFs) classifier [4].

3.1. PRLM

Figure 3 shows the block diagram of a PRLM system used to identify between two languages \( L_A \) and \( L_B \). During training, the training utterances for \( L_A \) are first converted into phone sequences using the phone recognizer for language \( L_1 \). The phone sequences are then used to train a phone language model for \( L_A \). The phone language model for \( L_B \) is built using the training utterances for \( L_B \) in a similar fashion. During recognition, an unknown utterance \( X \), is first run through the phone recognizer, which outputs the phone sequence \( O \):

\[
O = \text{argmax}_{O'} P(O' \mid X)
\]  

where \( P(O' \mid X) \) is the confidence score of the phone recognizer in generating the phone sequence \( O' \) given \( X \). Next, the log likelihoods for \( O \) under the trained phone language models for \( L_A \) and \( L_B \) respectively were calculated. \( \hat{l} \), the hypothesized language of the utterance, is chosen as:

\[
\hat{l} = \text{argmax}_{l \in \{L_A, L_B\}} \log P(O \mid \lambda_l)
\]  

where \( P(O \mid \lambda_l) \) is the likelihood for \( O \) under the phone language model for language \( l \). Note that in PRLM the language of the phone recognizer can be different from the languages to be identified, i.e. \( L_1 \) need not be the same as \( L_A \) or \( L_B \).

3.2. Parallel PRLM (PPRLM)

PPRLM is a commonly used variant of PRLM, which has been shown to yield better performance than a single PRLM [3]. In a PPRLM system, multiple PRLM systems are run in parallel with each PRLM system using a phone recognizer for a different language. The results from each PRLM system are then combined together to produce the final result.

Figure 4 shows the block diagram for the PPRLM system that we used, which contains two phone recognizers for languages \( L_1 \) and \( L_2 \) respectively. The training process is similar to that for PRLM, the difference being that two pairs of language models are built instead of one. During recognition, the unknown utterance is run through each PRLM, producing two phone sequences \( O_{L_1} \) and \( O_{L_2} \). The final likelihood scores for the utterance under each language are calculated as the sum of the individual log likelihoods from the corresponding language models associated with the language.

\[
\hat{l} = \text{argmax}_{l \in \{L_1, L_2\}} \left( \log P(O_{L_1} \mid \lambda_{L_1}) + \log P(O_{L_2} \mid \lambda_{L_2}) \right)
\]
### 3.3. PPRLM with CRF classifier (PPRLM+CRF)

We propose an extension to PPRLM wherein the individual confidence scores of the phones in the phone sequences as output by the phone recognizers are integrated during language identification. We replace the decision equation (3) in PPRLM by a CRF classifier and cast the LID problem as a sequence labeling problem. The phone sequences \( O_1 \) and \( O_2 \) are first aligned monotonically using dynamic programming, and then in addition to each pair of aligned phones we include information derived from the phones’ confidence scores and the overall language model scores. The CRF classifier then attempts to label this observation sequence with the language identity. During training, the label of each observation sequence is given as the language of the utterance from which the sequence is generated. During testing, after labeling every observation sequence, the language identity of the entire sequence is generated. The open-source CRF++ toolkit\(^1\) was used to implement the CRF classifier.

The features used within the CRF model are as follows:

1. \( P_{n,L}(\cdot)\): This is a feature derived from the difference between \( C_{n,L} \) and \( C_{n,L+1} \), where \( C_{n,L} \) denotes the confidence score (log likelihood) assigned to the phone \( P_n \) by the phone recognizer. This feature takes the value ‘+’ if the difference is positive, ‘-’ if the difference is negative, and ‘=’ if the difference is zero.

2. S-C-Delta\(_n\): This is a feature derived from the difference between \( L_{M_K} \) and \( L_{M_A} \) in a fashion similar to S-C-Delta, where \( L_{M_K} \) denotes the confidence score (log likelihood) assigned to the phone \( P_n \) by the phone recognizer. The observation sequence is formed with the feature set \( \{P_{2,1}, P_{4,1}, P_{4,0}, P_{1,1}, P_{2,2}, P_{2,2}, P_{1,2}, P_{1,2}, P_{2,2}, S-C-Delta_2, S-C-Delta_1, S-C-Delta_0, S-C-Delta_1, S-C-Delta_2, S-LM-Delta\} \).

3. S-LM-Delta: This is a feature derived from the difference between \( L_{M_A} \) and \( L_{M_B} \) in a fashion similar to S-C-Delta, where \( L_{M_K} \) denotes the confidence score (log likelihood) assigned to the phone \( P_n \) by the phone recognizer. The observation sequence is formed with the feature set \( \{P_{2,1}, P_{4,1}, P_{4,0}, P_{1,1}, P_{2,2}, P_{2,2}, P_{1,2}, P_{1,2}, P_{2,2}, S-C-Delta_2, S-C-Delta_1, S-C-Delta_0, S-C-Delta_1, S-C-Delta_2, S-LM-Delta\} \).

In addition, the bigram feature in CRF++ is turned on.

### 4. Experiment Setup

We evaluated the classification accuracies of PRLM, PPRLM and PPRLM+CRF on the identification of English and Iraqi-Arabic.

The corpus used was the training corpus for TransTAC. We selected approximately twenty-eight hours of English and Iraqi discourse consisting of fourteen hours each of English and Iraqi utterances. For each language the first ten hours of the utterances were set aside for training, with the remaining four hours used for testing. In addition, evaluation was conducted on two common TransTAC evaluation data sets:

- June08-OPEN: Human mediated dialogue between English and Iraqi speakers.
- June08-NAMES: Information gathering dialogue. 30% of the English utterances contain mention of Arabic names, and 80% of the Iraqi utterances contain mention of names.

Table 1 contains the details about the lengths of the utterances in the data sets.

In total we evaluated seven systems with the configurations listed in Table 2. Existing phone recognizers for English, Arabic and Chinese from the GlobalPhone project \([5]\) were used without any modification. In all the systems the language model used was the trigram n-gram language model trained on TransTAC-Train, built with the SRI Language Modeling Toolkit (SRILM) \([6]\) using the default settings.

### 5. Results and Analysis

The classification accuracies of all the systems are shown in figure 5. First comparing among the PRLM systems, we observed that PRLM-AR and PRLM-EN performed better

---

1. \url{http://crfpp.sourceforge.net}
As we are interested in using LID in real-time, we measured the elapsed time taken for running PPRLM-AR-EN and PPRLM-AR-EN+CRF. The tests were conducted on a dual Intel Xeon 3.60Ghz server, and in both systems, the Arabic and English recognizers were run in parallel. The time taken in real time factor for PPRLM-AR-EN is 0.103. For PPRLM-AR-EN+CRF, though it incurs an extra cost in using CRF classification, the increase in time taken in real time factor is only 1%.

6. Conclusions

We described a framework that enables LID capability to be incorporated into real-time S2S translation systems while minimizing the impact on the responsiveness of the systems. We evaluated the PRLM and PPRLM approaches on the identification of English and Iraqi-Arabic, and also propose an extension that uses a CRF classifier and allows for the inclusion of individual phones’ confidence scores. The extend approach improves the accuracy over that of the basic PPRLM approach by an average of 2.1%.

In the future, we intend to evaluate the usability of S2S translation systems with LID enabled. We also expect that incorporating discourse or dialog modeling into the LID process will increase the accuracy of identification, or help constraint the set of possible languages that needs to be identified, thus reducing the time required for LID. We will also evaluate other lower cost LID approaches such as GMM-based techniques.

7. Acknowledgment

We are grateful to Qin Jin of the Interactive Systems Laboratories at CMU for providing access to the GlobalPhone phone recognizers and for her technical assistance.

8. References