

An Analysis on Challenges & Strategies in Developing Multilingual Software for Mobile Technologies

Ganesh Babun¹ Dr. Mahammad. K. B.²

¹Research Scholar, CMJ University, Shillong, Meghalaya

²Professor of Computer Science, CMJ University, Shillong, Meghalaya

Abstract - *This paper describes experiences in designing and developing the software applications Indic languages targeted at mobile devices at CDAC Noida. CDAC has vast experience in multilingual computing and has also implemented several projects using browser based architecture. Now as we go further there is increasing trend towards browsing web through mobile devices in the remote places and while on the go, the Indic languages need to be tested out on various browsers for the performance. As a start a bilingual dictionary developed for different target platforms was ported on different mobile devices and experience was good as it is not easy for a developer to take off and start developing immediately. The environment requires lots of settings to successfully port and run application which have some language specific module. A lot of work needs to be carried out to make successful migration of useful applications and Indic languages websites accessible for masses on mobile devices.*

We evaluate different architectures to recognize multilingual speech for real-time mobile applications. In particular, we show that combining the results of several recognizers greatly outperforms other solutions such as training a single large multilingual system or using an explicit language identification system to select the appropriate recognizer. Experiments are conducted on a trilingual English-French-Mandarin mobile speech task. The data set includes Google searches, Maps queries, as well as more general inputs such as email and short message dictation. Without pre-specifying the input language, the combined system achieves comparable accuracy to that of the monolingual systems when the input language is known. The combined system is also roughly 5% absolute better than an explicit language identification approach, and 10% better than a single large multilingual system.

The coexistence of Western and Eastern languages and cultures poses a true challenge for global mobility and communication in business and personal life. In this paper, we will describe mobile software applications that are built on top of multilingual and cross lingual technologies for overcoming language barriers, e.g., between English and Chinese, or German and Chinese. The mobile applications are interactive language guides, culture guides and country guides. They support automatic translation and dialogues between people and link the language part with information needed in a specific situation. Furthermore, users can integrate pictures into their phrases for extending the descriptive capabilities of the dialogue component. Semantic search facilitates access to words, phrases and information content.

INTRODUCTION

India is the best example of multilingual country in the world. There are 22 constitutionally recognized languages in India. In the last few years, there has been a sea change in the technology with global economy and strong

presence of Indian companies in IT market. As the economy progressed the change has been seen in the lifestyle of people and information and communication has become an important aspect. This fact has evidence in reports published recently in the national newspaper Hindustan Times that "India, the world's fastest growing

wireless services market, added 6.09 million new mobile customers in September 2006, boosting its user base to 129.53 million. Of the new additions, 4.39 million opted for services that run on the widely prevalent GSM platform, while 1.7 million signed up for CDMA-based services. The total number of new users in September was higher than the 5.9 million subscribers, who joined the booming sector in August, 2006 and the 5.39 million that entered the market in July 2006.

With the increasing use of mobile phones, people are expecting more utility applications. This kind of user base will naturally attract companies across the world to expand their market base and establish their products. By some estimates, more than half of the world's population is multilingual, however most commercial recognition systems remain monolingual. At the same time, speech recognition is now being used both to get information from machines (e.g. speak a Google query) but increasingly to communicate with people by dictating short messages. Together there is increased pressure to recognize whatever language might be most appropriate for whatever setting, without requiring the user to navigate language-selection interfaces, which themselves are complicated by keyboard requirements and other geographic considerations. An omnilingual or at least multilingual recognizer would make many of these interactions more natural, but few users would choose that trade-off if accuracy or latency were degraded.

With those constraints in mind, we evaluated several multilingual techniques on datasets representative of our current mobile traffic, which is a mix of Voice Input (usually short dictation), Voice Search (Google queries), and Voice Actions (commands). These are mostly short utterances recognized in real-time. We started with a trilingual English-French-Mandarin scenario, and evaluated each technique on a union of three test sets representative of those languages.

One ambitious but conceptually straightforward application of advanced language technology is the combination of speech recognition and automatic translation for unrestricted dialogue translation. The first products of this type have entered the market. The technologically most advanced example is the app Jibbiggo. The product delivers amazing results, even in its bidirectional English/Chinese version. Nevertheless, the error rate is still too high for secure dialogues in critical situations, because neither speech recognition nor machine translation is mature enough for large-coverage reliable applications. Errors of both technologies add up and often combine into unpredictable and inexplicable out-put.

SINGLE MULTILINGUAL SYSTEM

One direct technique for multilingual recognition is to train a 'universal' acoustic model, capable of recognizing all (relevant) languages. This approach holds the promise of helpful data sharing between languages, and has been explored in various ways by many researchers, including us. It is also attractive for its ease of maintainability (one model for all languages), but might require fundamental decoder changes to accommodate large models while maintaining low-latency characteristics.

The mixed model we report on here ('Mix' below) was trained by merging the training sets and phone sets of all three languages (119 phones total). Pronunciations for each training word were extracted from the corresponding lexicon (or pronunciation engine).

Words appearing in several languages have pronunciations in several languages. This results in an average of 1.3 pronunciations per word, which is in line with the monolingual lexicons. The model contains roughly 900K Gaussians, which is a little under the sum of the number of Gaussians of the individual monolingual systems. It runs roughly 2 times slower than the monolingual systems. Its accuracy is summarized in Table 1.

System	SACC (%)			
	En	Fr	Zh (Zh/En/Mix)	Avg
Mix	41.6	39.8	32.6 (31.8/42.6/26.4)	38.0

Table 1. Mix system accuracy.

Despite our best attempts, we found it difficult to obtain high accuracies with a single model approach : the overall accuracy is more than 10% worse than the monolingual baseline (38 vs 49.5%).

Again, this is a data-saturated environment, so pooling resources does not compensate for data sparsity, and does not readily provide accuracy benefits. The exception is the English subset of the Mandarin test set, which largely benefited from the added American English training data.

YOUR MULTILINGUAL APP: GETTING STARTED

Categorize and organize your content : By categorizing the app content and setting priorities for what is most important to translate, you can make cost-effective translation decisions. If you are budget-constrained, you may have to make trade-offs between translating content and core app functionality.

Determine what content and functionality to translate : What functionality will you translate to provide an optimal experience for your users in every language? Typically,

this will include the user interface elements like menus, button labels, settings, and in-line help. When you translate, you also need to localize items like dates, times, measures, and currency. You may also choose to localize some design elements, such as colors and images.

For many apps, choosing what content to translate is fairly straightforward, e.g., a tourist information app might need to translate locations and descriptions for popular destinations.

Some content may be displayed in an app but generated by a web server. In an e-commerce application, for example, the app may pull product information from the store website.

For the user to see that information in the language of the user interface, you'll need to translate the content of the server application as well as the mobile app. Pricing and currency from the server application will need to be localized.

Figure out how to deal with User-Generated Content (UGC) : There are significant business decisions required when a mobile app includes UGC such as user reviews and ratings. Should the UGC be translated, or should the app display it only in the user's language? Regardless of what you decide to translate, you will want to ensure that your app accepts user-generated text in any language, independent of the language of the user interface. The same applies for server side applications that work with the mobile app.

Try to plan ahead for translation when writing copy in the app's "original" language. Some phrases are difficult to translate (slogans and jargon are notorious for causing problems). It's a good idea to consult with a professional translator or locale-specific marketing professionals to determine how best to represent your brand.

Select the translation method that fits your business and budget : You can translate your app using professional translation, crowd sourcing, or machine translation. If precision and consistency in word choice are essential to the functioning of your app, you may need to work with professional translators. Crowd sourcing is a smart option for apps with active early adopters who speak the target language; you may be able to engage them to translate content, with an expert coordinating the work. For some businesses, machine translation may be sufficient, or you may decide to use a combination. Whichever path you choose, it's a key decision.

Categorize and organize your content : By categorizing the app content and setting priorities for what is most important

to translate, you can make cost-effective translation decisions. If you are budget-constrained, you may have to make trade-offs between translating content and core app functionality. Determine what content and functionality to translate What functionality will you translate to provide an optimal experience for your users in every language? Typically, this will include the user interface elements like menus, button labels.

CHALLENGES

Testing of Applications : Testing is a critical part of any software development life cycle and in case of localized applications it becomes even more important and complicated when application requires different combinations of hardware and testing scenarios. It requires extensive quality assurance testing, both functionally and linguistically.

- Functional testing of localized applications involves mix of hardware and multilingual environments. Functional testing can be completed with the actual devices or emulators. If the application runs on a simulator, it should be tested using various target locales in mind. Linguistic testing is the performance in functionality and usability. The testing and quality assurance people should have knowledge of targeted language to report problems related to rendering and display of characters, meanings conveyed through translations, handling composite messages and usability.

- There are number of options for mobile devices in India but it has been observed that there is huge market for low-end devices. Handsets manufacturers generally tie up with the Service providers to popularize their phones for wide reach using promotional schemes. The applications therefore need to be tested for different handsets having different display sizes and screen resolution apart from different operating environment.

- Developers of websites and tool vendors for mobile devices sometimes forget to take into account the constraints such as rigid layout, small size of screen, resolution, navigation links, large pages to be loaded over cellular network, application in need of large memory and processing power.

- The mobile browser environment is highly fragmented. There are a number of browsers available for mobile devices. This situation has resulted in a number of challenges:

- o Tools vendors are yet to address mobile browsing in their tools;

o Content developers have not really focused on mobile browsing as a target segment from Indic languages point of view. How the text should be encoded for mobiles UNICODE (UTF-8) or some other encoding.

o Availability of generic and language specific guidelines for content developers in Indic languages.

CONCLUSION

The most promising approach is to allow utterances to be recognized in parallel by several systems, and combine the scores of these systems with a classifier. A simple confidence voting scheme between three monolingual systems brought us closer to monolingual accuracy than any system we previously evaluated, and adding other easily-computed knowledge sources such as language ID scores helped bridge most of remaining accuracy gap.

The future of the mobile interface is the Internet. As of now, mobile browsing is still in infancy in India, but few years down the line, most users will connect to Internet and use a mobile browser who are currently accessing it through desktop. This emphasizes bringing improvements in mobile browsing experience for users by offering fast speed, useful services and good usability. It is true that mobile phones are outsourcing fixed lines in terms of number of users but in India we have long way to go to make use of mobile devices for utilities of mass use, as people are still using low-end phones and are getting acquainted with the technological advancements. The mobile even today is being used widely as a medium of communication. However as the technology has been advancing rapidly and smart devices are becoming cheaper day by day, soon ,people will start taking the advantage of these devices for uses other than the vocal conversation. We have to really make ourselves ready to take these advancements ahead by testing the development tools for applications for mobile. Fonts and encoding issues should be resolved by choosing uniform encoding for exchange of information across all the devices and making available Indic language fonts for all devices and platforms. Guidelines for developing the content for browsing through mobile devices and usability standards should be evolved.

REFERENCES

- C. Alberti, M. Bacchiani, "Discriminative Features for Language Identification", Interspeech 2011.
- D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, K. Visweswariah, "Boosted MMI for model and featurespace discriminative training", ICASSP 2008.
- G. R. Tucker, "A Global Perspective on Bilingualism and Bilingual Education", CMU, 1999.
- H.A. Chang, Y.H. Sung, B. Strophe, F. Beaufays, "Recognizing English Queries in Mandarin Voice Search", ICASSP 2011.
- J. Kohker, "Language adaptation of multilingual phone models for vocabulary independent speech recognition tasks", ICASSP 1998.
- J. Shan, G. Wu, Z. Hu, X. Tang, M. Jansche, P. Moreno, "Search by Voice in Mandarin Chinese", Interspeech 2010.
- J.T. Huang, H. Lin, Y.H. Sung, B. Strophe, F. Beaufays, "System Combination to Recognize Mandarin and Accented English", ICASSP 2012.
- M. Gales, "Semi-Tied Covariance Matrices for Hidden Markov Models", IEEE Trans. SAP, May 2000.
- N.T. Vu, F. Kraus, T. Schultz, "Rapid building of an ASR system for Under-Resourced Languages based on Multilingual Unsupervised Training", Interspeech 2011.
- T. Schultz, A. Waibel, "Language Independent and Language Adaptive Large Vocabulary Speech Recognition", Speech Communication, Vol. 35, 2001.
- W. Campbell, J. Campbell, D. Reynolds, E. Singer, P. Torres-Carrasquillo, "Support vector machines for speaker and language recognition", Computer Speech & Language, vol. 20, no. 2-3, 2006.