TransSMS: A Multi-Lingual SMS Tool

Mazliza Othman, Bikesh Lakhmichand
Faculty of Computer Science & Information Technology
University of Malaya
50603 Kuala Lumpur
Malaysia

Abstract

TransSMS is a multilingual SMS solution developed under the Intercultural Collaboration Experiment 2002 (ICE 2002). It provides features that enable users to send and receive text messages in different languages. At the moment, the languages supported are Malay, Japanese, Korean, Chinese and English. TransSMS provides two main features: (1) translating a text message from one language to another and sending the translated text as SMS; (2) 'read' the translated text to the user. The second feature is useful for tourists who do not speak the language of the country that he is visiting. TransSMS can be accessed via the Web or a Java-enabled phone.

1 Background

The Intercultural Collaboration Experiment 2002 (ICE 2002) [1] is an experiment headed by the Department of Social Informatics, Kyoto University, **Japan**, as an initiative to conduct an experiment on open source software development by multi-national teams in Asia. The participants of the experiment are the Faculty of Computer Science & IT, University of Malaya, **Malaysia**, the Department of Computer Science & Engineering, Shanghai Jiaotong University, **China**, and the Department of Management & Computer Science, Handong University, **Korea**. Because one of the unique characteristics of this experimental project is the pursuit of collaboration among heterogeneous groups across country borders, English was not used as a standard language. One of the objectives of the experiment is to see if the participants are able to break the language barrier through the use of machine translation tools.

Two tools were provided by Kyoto University at the beginning of the experiment, i.e. TransBBS and TransWeb. *TransWeb* is a tool that allows participants to read all Web pages in the their native languages by specifying a URL and selecting a language. All web pages are then translated to the participant's native language. *TransBBS* is a Bulletin Board System that incorporates multi-lingual translation. It provides support for Malay, Chinese, English, Japanese and Korean languages, thus, allowing participants in different countries to communicate with each other in their native languages. Each participant can choose a language when viewing messages. Participants are required to use their own native languages to discuss software design during the experiment. During the first track of the experiment, the four teams worked together to add new functions to these tools.

The China team improvised the functionalities of TransBBS to create the TransSearch engine that translates the query term into either of the 5 languages. The search results is translated into the selected language. The Japan team developed TransMail, a web-based mail program to send messages among team members instead of using the bulletin board provided by TransBBS. The Korea team devised a chat program called TransChat which functions like a normal chat program with capabilities to converse in multiple languages simultaneously. The Malaysia team developed TransSMS, a multilingual SMS solution that enables users from the 4 nations to send and receive messages in different languages. At the moment, the languages supported are Malay, Japanese, Korean, Chinese and English

2 TransSMS

The TransSMS service can be accessed via the Web or a Java enabled phone that has already downloaded the TransSMS client software. There is no difference in terms of functionality between the two methods. Both include security features and text to speech translation capability. The analysis diagram in Figure 1 gives an overview of the modules within the system and how they interrelate.

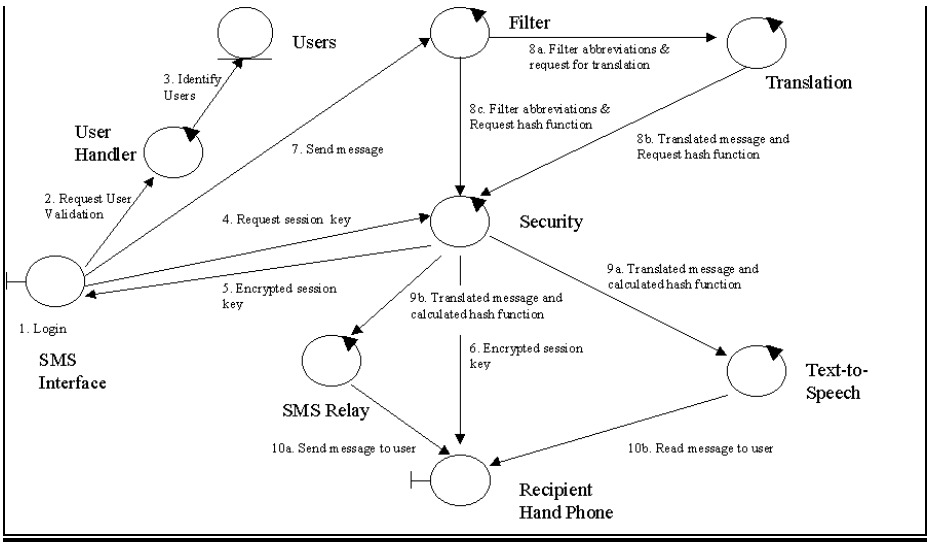

Figure 1: *Analysis Diagram of TransSMS*

The user may request for the translated text to be sent as SMS to a recipient or request for a Call Back. In case a Call Back is requested, the service center calls the sender or recipient and read the translation using a text to speech engine.

The login is only used for the web client to easily track usage for billing purposes. An option is provided to ensure that messages are not tampered with during transfer includes a hash function and encrypted session key generated to the user and recipient.

A *filter module* parses the message and looks for abbreviations by matching words against an abbreviation database. If there is a match, the abbreviation is replaced with the appropriate word. Even though currently only English-based abbreviations are supported, the algorithm will work for other languages as well if abbreviations for that language are keyed into the database.

The system can also recognise emoticons, such as " :)", and replaces it with its corresponding meaning (i.e. Smiley Face, Giggle). This substitution is only needed during text to speech processing as those specific symbols would not make any sense when spoken as "colon parenthesis". The text to speech engine would process the sentence as naturally as possible by either inserting a "Hahaha" or describing the emoticon as a "Smiley Face" both depending on how the emoticon database is defined. The database holds all standard emoticons used online and their appropriate representations. On the other hand, SMS translations leaves the emoticons as it is since as they are understood globally.

The translation process involves the system parsing the sent data to identify the original message language, the desired translated language and result format. If the requested result is an SMS, then the message is translated appropriately and relayed to the recipient's number. If the result is a Call Back, the server will translate the result, place a call to the user and read the translation to the sender.

3 TransSMS Web Client

Figure 2 shows the main screen of the web-based TransSMS. The network option allows the user to select a supported network and helps ensures that the telephone number is in the correct format. The telephone number consists of the country code + network number + hand phone number, e.g. to call a Maxis line 302 7527 in Malaysia, you would key in +60123027527.

The message language dropdown boxes allow user to specify the language of the original message entered in the message area and the language it is to be translated to. Figure 3 shows an example of a message in English that is to be translated to Chinese. Upon clicking the 'Translate' button, the message is translated into Chinese as shown in Figure 4. Clicking the Send button leads the user to a confirmation screen followed by the actual sending of the message to the desired mobile number. The read option allows the user to listen to the text spoken in the desired translated language. The server processes the text as speech and streams it to the user in the form of a wave file.

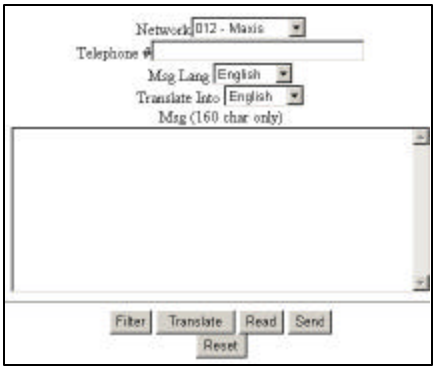

Figure 2: *TransSMS Main Screen*

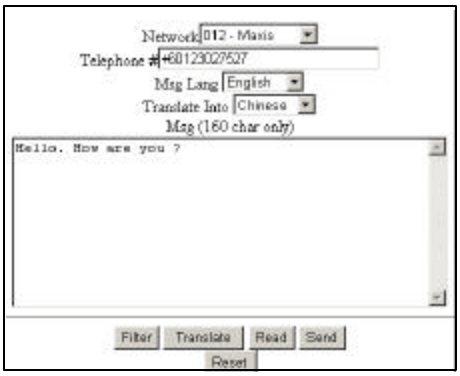

Figure 3: *Message to be translated*

Figure 4: *Translated message*

4 TransSMS Mobile Client

The mobile client version is written using the micro edition of Java (J2ME). Users with web access on their mobile devices can navigate the main TransSMS site to download the TransSMS clients into their devices. In the current version of the client, it utilizes the SMS technology to transmit to the server the required parameters. This can be modified to use web access to transmit the data for users with existing General Packet Radio Service (GPRS) connections or other various connections.

Settings within the client involves setting the service centre number, which is a special telephone number provided by the service centre to process SMS translation requests. If a GPRS-based method is adopted, the service centre IP address is specified instead for the messages to be sent to it.

Sending a message via the mobile client is similar to sending a regular SMS. The user selects the 'Compose' menu and proceeds to type in a message (see Figure 5 and Figure 6 respectively). Once the message is ready, the user selects the 'Options' command and proceeds to send the message by entering the recipient's telephone number. The user can also change the format of the message from the compose screen should he need to change the desired translated language.

Figure 5: *Main Menu*

Figure 6: *Composing a message*

Figure 7: *Options Menu*

Figure 8: *Recipient's telephone number*

Figure 9: *Send confirmation*

Figure 10: *Message Settings*

Figure 11: *Language setting*

Figure 12: *Selecting a language*

Figure 13: *Result Format*

Figure 14: *Selecting result format*

The message format settings (Figure 11) allows the user to select the message language and the desired translated language. This option can be accessed via the Settings menu and also while composing a message. If the translation setting is not often changed, a predefined translation pair can be initially set. Should the user need to modify from the current setting for a particular message, the option can be easily accessed while composing the message. Another setting provided is the result format setting that determines the format to the server 's reply, i.e. as a translated SMS or a Call Back.

5 TransSMS Server

The server may connect to a mobile phone either by an infrared or a serial cable connection. A Component Object Model (COM) is used to process incoming SMS. Each SMS is parsed in order to filter the abbreviations and to translate the messages. If the result format requested is SMS, the translated message is relayed to the appropriate recipient. This is done either by using the connected mobile phone to send an SMS or using an SMS web service.

The Telephony Application Programming Interface (TAPI) [3] and Speech Application Programming Interface (SAPI) [2] are two standards that can be used of when developing voice telephony applications. TAPI is a standard program interface that lets the computer "talk" over telephones or video phones to people or phone-connected resources elsewhere in the world. SAPI is an application program interface (API) provided with the Microsoft Windows operating system that allows programmers to write programs that offer text-to-speech and speech recognition capabilities. Interfaces are provided for the C, C++, and Visual Basic programming languages. Using Microsoft's COM architecture, SAPI is the most widely used speech application program interface used today.

If a Call Back format via a web is requested (basic read command), a component utilizing Microsoft's Speech Application Programming Interface (SAPI) will speak the translation into a wave file. The user is then prompted to download and play the file in order to listen to the translation. This method ensures the speech content is accessible by all users and need not require any special SAPI plug-ins installed. A Call Back request from a mobile device requires the server to call the sender's phone via Microsoft's Telephony Application Programming Interface (TAPI) and read the translation to the user using SAPI.

The streaming of the wave file to users via the web is a multithreaded process ensuring that the server can handle simultaneous requests for various translations. The Call Back feature for mobile devices is initially designed to work as a queue. Requests are queued and dialed one at a time as the current server setup only involves a single phone connection.

6 Conclusion

The main objective of this project is to develop an enhanced SMS service that can be used by telecommunication companies in the four countries to provide a value–added service to customers. Eventually, this technology can be used by telcos all around the world to improve the quality of their service as well as increase the number subscribers. The service will give an upper hand to telcos that have a vision of bringing the citizens of the world closer together.

References

[1] Saeko Nomura, Toru Ishida, Mika Yasuoka, Naomi Yamashita, Kaname Funakoshi, *Open Source Software Development in Your Mother Language: Intercultural Collaboration Experiment 2002*, to appear in the Proc. of HCI International 2003, 22 – 27 June, 2003, Crete, Greece.

[2] Speech Application Programming Interface, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcesapi/htm/ceconsapi50overview .asp

[3] Telephone Application Programming Interface, http://msdn.microsoft.com/library/default.asp? url=/library/en-us/tapi/tapi3/tapi_3_1_start_page.asp