Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

Om: One tool for many (Indian) languages

GANAPATHIRAJU Madhavi[†1], BALAKRISHNAN Mini[2], BALAKRISHNAN N.[†2], REDDY Raj[†1]

([1]*Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA*)

([2]*Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore 560 012, India*)

[†]E-mail: madhavi@cs.cmu.edu; balki@serc.iisc.ernet.in; rr@cmu.edu

Received Aug. 5, 2005; revision accepted Sept. 10, 2005

Abstract: Many different languages are spoken in India, each language being the mother tongue of tens of millions of people. While the languages and scripts are distinct from each other, the grammar and the alphabet are similar to a large extent. One common feature is that all the Indian languages are phonetic in nature. In this paper we describe the development of a transliteration scheme Om which exploits this phonetic nature of the alphabet. Om uses ASCII characters to represent Indian language alphabets, and thus can be read directly in English, by a large number of users who cannot read script in other Indian languages than their mother tongue. It is also useful in computer applications where local language tools such as email and chat are not yet available. Another significant contribution presented in this paper is the development of a text editor for Indian languages that integrates the Om input for many Indian languages into a word processor such as Microsoft WinWord®. The text editor is also developed on Java® platform that can run on Unix machines as well. We propose this transliteration scheme as a possible standard for Indian language transliteration and keyboard entry.

Key words: Om transliteration, Indian language technologies, Text editor
doi:10.1631/jzus.2005.A1348 **Document code:** A **CLC number:** TP391

INTRODUCTION

India is a nation with pluralistic culture, a large number of cultures, ethnicities, languages and religions coexisting with each other. While the culture and faith unify the country under one umbrella either by similarity or by tolerance, the language is what separates them. In the 1951 census, the first census after India attained independence, 845 languages (dialects) were identified, of which 60 were spoken by at least 100000 people each. The Indian constitution identifies 22 languages, of which six languages (Hindi, Telugu, Tamil, Bengali, Marathi and Gujarati) are spoken by at least 50 million people within the boundaries of the country—there are a large number of them living outside the country. Although the Indian languages were identified as belonging only to four different language families, namely, the Austric, Dravidian, Tibeto-Burman, and Indo-Aryan, the language spoken by one person is rarely understood by a person familiar only with another language; this does not however rule out bilingualism of a large number of people, especially those who migrate from one state to another, where they speak the mother tongue at home and can usually follow the dominant language of the new state. For example, Telugu speakers are found in good numbers in Karnataka (3 325 062), Maharashtra (1 122 332), Orissa (665 001), and Tamil Nadu (3 975 561); about 10% of Telugu speakers live outside of the Telugu territory, according to an old 1901 estimate; this number would be much larger today. Bilingualism is also found at the borders of two states, where people can usually speak languages of both the states sharing the border. Taking the example of Andhra Pradesh again, where the native language is Telugu, a large number of people speak languages of its neighbours: Kannada (519 507), Marathi (503 609), Oriya (259 947), and Tamil (753 484).

Language technologies and PC penetration in India

India is fast becoming a software superpower—the nation has over 3000 computer training institutes; software exports were about 6 billion US dollars in 2003, and are expected to grow to 50 billion US dollars, which is 33% of total exports, very soon. However, net-surfers that were at 0.2% of the total population are expected to grow only up to 7% by 2006. The PC penetration rate is merely 1.4%. Sixty-eight million homes out of 408 million homes (17%) in the country have a TV, while only 22 million (5%) have a telephone; which is still much larger compared to the 1.4% penetration of a computer. Two most important influencing factors for this low computer usage by non software-professionals may be low income and illiteracy. The low income population in the country, which is a third of the total population, prefers to buy a television set rather than a PC because of the entertainment value, ease of use and the current non-utility of a PC in their everyday life.

At the time of the birth of independent India, about half a century ago, the Indian middle class was an insignificant minority; although the middle class is upwardly mobile. With the economic reforms brought about in the early 1990's, the Indian middle class is growing at a rapid rate and is expected to reach 50% within a generation, and the poverty is expected to diminish to 15%. Complementing the economic growth rate, the new Indian middle class is filled with entrepreneurs who are spreading the power of information technology to the rural areas. Although the PC has not yet penetrated into rural homes, there are countless Internet facilities (called cyber-cafés) that are expanding similar in scope and impact to the public telephone booths in the rural areas. Low-end computers, costing about $100 to $200 are coming to the market (Simputer, Mobilis, Nova NetPC). Thus, irrespective of economic status, the power of information technology is expected to be available for the Indian population very soon.

The second limiting factor in PC usage, however, is non-availability of the operational software in native language, and the language barriers between people. While the development of an operating system in the native language is a solution, this is likely to be limited to only a couple of languages; and the development of natural language processing technologies would have to wait until the standardization of the digital representation; the porting of available scientific knowledge in the areas of natural language processing would face the bottleneck of a local expert in the native language. If the Indian language texts were instead available in parsable English-like texts, they would seem attractive to the international research community in language processing. Isolated development of digital representations for the different Indian languages may further widen the language barrier in the country.

Thus there is a need for the development of a digital representation that lays a common foundation for all the Indian languages. For seamless adaptation of algorithms in language technologies, this representation must also be parsable by universal language processing tools and algorithms, such as for machine translation, information retrieval, text summarization and statistical language modelling.

The representation must exploit the common alphabet of the various Indian languages. It must cater to the increasingly large number of people that can speak, but not read the native language—these people often can read another Indian language or English.

PRIOR TRANSLITERATION METHODS BUILT AROUND STANDARD KEYBOARD

ITRANS is a representation of Indian language alphabet in terms of ASCII (http://www.aczoom.com//itrans/). There are typically about 13~18 vowels and 36~54 consonants in the Indian language—while there are only 26 letters in the English alphabet. Since Indian text is composed of syllabic units rather than individual alphabetic letters, ITRANS uses a combination of two or more letters of the English alphabet to represent an Indian language syllable. However, there being multiple sounds in Indian languages corresponding to the same English letter, not all Indian syllables can be represented by logical combinations of the English alphabet. Hence, ITRANS uses non-alphabetic characters such as "[", "\", "'" in some of the syllables. These combinations are not logical and are not easy to remember and recall. ITRANS notation is also case dependent—it uses capital and small letters to represent different language syllables. Another major drawback of ITRANS proposed so far is that the same Indian language character can be rep-

resented in more than one way using lower and uppercase letters, making the transliterated non-uniform across people.

Unicode standardization captures the commonality in the alphabet of various Indian languages, but does provide an input mechanism. It does not provide a logical mechanism of applying the language parsing algorithms on texts encoded in this format. The lexical ordering of the Indian languages cannot be applied in a logical fashion. The representation does not automatically transliterate to English, which is an important requirement as discussed above.

A very significant contribution in this area is that of the Acharya group at the Indian Institute of Technology, Madras (http://acharya.iitm.ac.in/). They have developed a representation that preserves the syllabic and phonetic nature of Indian languages and also preserves lexical ordering. However, the representation is only machine-readable, but the input and English transliteration are still based on ITRANS. This is very good for internal representation and also for lexical ordering and syllabic parsing such as finding palindromes in the text. But absence of a mapping to ASCII makes it in-adaptable to standard language parsing applications.

To overcome the drawbacks of ITRANS we have redesigned a novel mapping scheme called OM, which is no longer a transliteration mechanism alone, but a platform over which many other Indian language applications have been built; the details of which are described in the rest of this paper.

OM TRANSLITERATION: UNIFIED REPRESENTATION FOR INDIAN LANGUAGES

OM uses the same representation both for keyboard input and formation and representation. It is similar to ITRANS in that it uses combinations of the English alphabet to represent Indian syllables. However, it is case independent, and avoids excessive use of non alphabetic characters; where used they are consistent. Further, the English alphabet combinations are designed such that they are easy to remember at the time of input with standard keyboard and also natural to read like English. The case independent representation allows use of sentence and title case writing in a natural fashion; further, the texts are more highly readable than their ITRANS counterparts. It may be seen from any ITRANS text that the large mixture of capital and small letters, and not an alphabetic characters leave it highly difficult to read.

OM's features enhance the usability and readability, it has been designed on the following principles: (1) easy readability (2) case-insensitive mapping: while preserving readability, this feature allows the use of standard natural language processing tools for parsing and information retrieval to be directly applied to the Indian language texts and (3) phonetic mapping, as much as possible: this makes it easier for the user to remember the key combinations for different Indian characters ASCII representation may be used simply as a means of typing the text with standard keyboard. (4) OM separates the storage that is in ASCII and the rendering that is dependent on the fonts chosen. This paves the way for a language independent universal representation; a fact that had been exploited in multilingual search engines (Jayaraman *et al*., 2004). For transliteration to Indian languages, OM representation is mapped to the Indian language fonts for display or converted to any other format such as Unicode or Acharya, where required. When a user is not interested in installing language components, or when the user cannot read native language script, the text may be read in English transliteration itself. India being a multi-lingual country, and inter-mixed population, the people can often speak and understand more than one Indian language and also English. Hence even in the absence of OM to native font converters, people around the globe can type and publish texts in OM scheme which can be read and understood by many even when they cannot read native script. The readability criterion that is benefited from the case-insensitive phonetic mapping thus proves very useful.

The OM mapping tables for many Indian languages are shown at http://swati.dli.ernet.in/Om/. The table also shows the mapping for the characters, and some sample OM texts. Mapping table for Kannada is shown below as an example.

A fully-filled mapping table with OM characters as columns and different Indian languages as rows is also created (Fig.1): in this Figure, wherever a character present in one Indian language alphabet is not present in the alphabet of another, it is substituted with a similar sounding character in the latter lan-

guage. For example, the Tamil character *n-* is not present in Telugu, and hence the OM character *n-* is substituted with *n'* in Telugu.

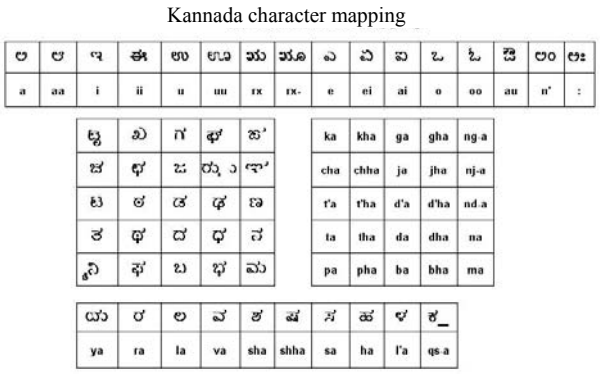

Fig.1 OM transliteration mapping for Kannada: The transliteration mapping table is shown for one of the south Indian languages, namely Kannada. As can be seen, the characters used are all lowercased letters, meaning that the transliteration is case-independent. This allows the use of sentence and first name capitalization in running text, without leading to any transliteration ambiguities. The extra characters used in transliteration are "'", "-" and ":", which are used in such combinations that are easy to remember. For example the character combination nd-a is designed based on how it is pronounced: the sound combining 'n' with 'd' with a '-' denoting they are part of the same character. Mapping tables for other languages may be seen on the website. The mapping above results in a quite well "readable" transliteration in English, compared to other transliteration schemes that use mixed cased letters

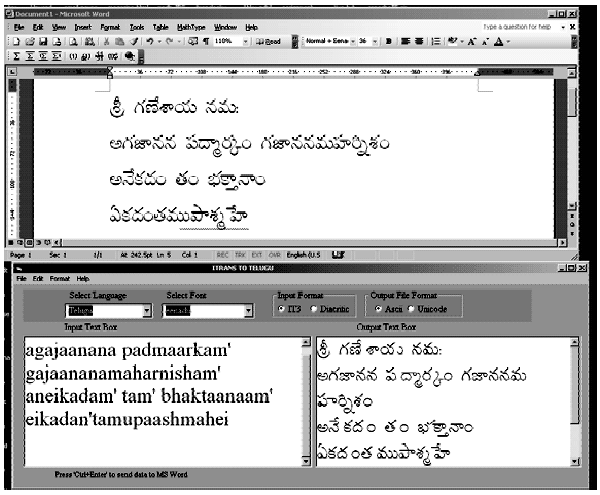

Fig.2 OM editor for Windows platform: The snapshot shows the multilanguage editor for text editing on the bottom, and Microsoft WinWord on top. The editor has two pull down menus that allow the user to choose the input language and the display font for that language. The output can be chosen to be ASCII (the chosen font) or Unicode. The ASCII input per OM transliteration can also be stored as a plain text file. The text entered by the user is shown in English characters on the left, and in native font on the right. The text on the right is updated for every keystroke by the user, allowing the user to correct any spelling mistakes by being able to see the output while typing. The native language text (in the right panel) may be sent to Microsoft WinWord for further formatting and printing by pressing "Control+Enter"

Word processor

An integrated transliteration package that accepts OM ASCII keystrokes as input and maps them to native fonts has been developed (Fig.2). The script in any one of the chosen true type fonts is sent to MS Word for further formatting and layout options. Since the OM scheme is common to all the Indian languages, the display of the text can be converted between the supported languages by choosing it on the menu. The text may also be saved in plain ASCII and Unicode formats. The tool also integrates with email clients on the Windows platform. A Web-interface with similar functionality has also been developed. The text may be saved as OM text, native font text or in Unicode. This does not support formatting explicitly but can be independently opened in MS Word like applications for such functionality.

Easy support for new languages

A mapping table between OM symbols and the glyphs of the font of the new language is required. Once this is provided, it is only a matter of a few minutes to integrate this new language into the package. All the other features of transliteration to other languages and use of word-editing features of Microsoft Word are available after the integration of the new font into the package. Currently, the OM transliteration package supports eight Indian languages.

Key in the input as we speak

The most notable feature of the OM transliteration package is that we can key in the input data just the way it sounds when we speak. For example if we have to key in 'Bharat' just type 'bhaarat'.

Uses lowercase English alphabets and some special characters

The use of lowercase letters provides awesome power to adapt language modelling tools such as

stemmer, translation, etc. The special characters used in OM are ‘, *, ~.

Switch between the languages at the click of a mouse

The option to choose any language and font is incorporated in the interface of OM by which switching from one language or font to the other is made easy.

Saves the output in ASCII and Unicode format

The file menu of the interface provides an option to save the input as well as the output, so that the user can import it later for future use.

Exchange email in Indian languages

This feature lets the user send electronic mail in plain text in Indian languages.

Integration with Microsoft® WinWord (MS Word)

The text generated in the native font by the OM text editor can be exported to Microsoft Word application (provided that the MS word application is present on the user’s computer). This means that all the features of MS Word, for example, format and layout management, printing, saving to other formats such as HTML, can now be used for Indian language text generated by the OM text editor.

Platform independent package in Java

The integrated editor that supports data entry with standard keyboard and transliteration of text between different Indian languages has also been developed on the Java platform, and can be run on any of the Linux or Unix machines. The text may be saved as OM text or native language font text (ASCII) or as Unicode text. The user interface is similar to that for the Windows platform, and is shown in Fig.3.

Web interface

For those who wish to create content using a Web interface, without the need to install the package locally, a Java based Web interface is also available (http://swati.dli.ernet.in/om/ and http://www.cs.cmu.edu/madhavi/Om) (Fig.4). The Web interface creates the output in plain text format, which may be opened in MSWord with the appropriate font selection, thereby using the full functionality of MSWord for the Indian language text editing.

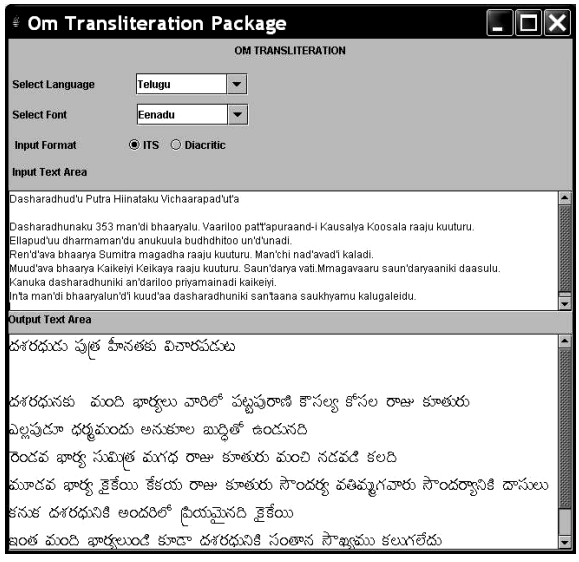

Fig.3 Java editor: A multilanguage text editor supporting OM keyboard entry and transliteration is shown in the snapshot. The features are the same as those for Windows based editor, except that this can run on Linux platform. Automatic export to WinWord is not supported; however, the native language font text can be stored in ASCII and it may be opened in any of the word processing applications for further formatting and printing

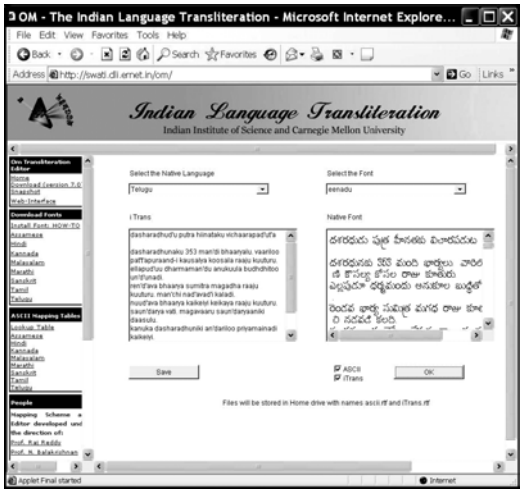

Fig.4 Web interface for OM transliteration: The Web interface developed on the lines of the integrated editor shown in Figs. 2 and 3 is shown here. The interface may be used to type text in OM transliteration in any of the supported Indian languages. The output rendered in native language with the chosen font is shown in the right panel updated on the fly for every keystroke. The output may be stored again as native language ASCII characters and may be opened in any of the text editors for further formatting and printing. Input may be saved as plain text file

CONCLUSION AND FUTURE WORK

A transliteration and keyboard entry scheme for Indian languages called OM has been described in this paper. Integrated text editing tools, for both Windows and Linux platforms, and also a Web service for the same, have been presented. The editor allows entry of text using OM mapping scheme using a standard keyboard, and converts the text to native language fonts. The editor, and the design of OM, also allow transliteration of the text from one Indian language to another. All the tools are freely available for use, downloading and hosting. Supplementary material consisting of all the mapping tables and inter-conversions between different languages is available on the website.

AVAILABILITY: FREE FOR DOWNLOADING AND HOSTING

The OM transliteration mapping and integrated editor can be downloaded at http://swati.dli.ernet.in/om/ and http://www.cs.cmu.edu/~madhavi/Om. The tools have been used extensively for data entry for texts that feed into applications such as machine translation and optical character recognition (Balakrishman *et al*., 2005). It has also been used purely for content creation by the outside community. An example may be seen at the magazine section of www.telugumn.org, where the story of Ramayanam has been created using this software. The integrated editor will also be provided for free of cost or use or hosting at any website, such as done at http://www.telugumn.org. The integrated editor is available for Windows and Linux platforms.

NATIONAL STANDARD

In India, the Ministry of Communication and Information Technology under its "Technology Development for Indian Languages" (TDIL) has been working on evolving national standards for representation and localization. http://tdil.mit.gov.in/homepage.asp describes some of the past and present attempts in standardization including the use and issues connected with the ISCII, UNICODE, INSFOC and INSROT. There have been many scattered attempts in this direction by some of the academic institutions and research organizations across the country. It is proposed to have a national conference of all the language and computer experts to brainstorm and decide on evolving an acceptable national standard like OM so that in all our future endeavors language as a barrier to ICT applications reaching the Indian rural populations and to the success of our E-Governance exercises would remove.

ACKNOWLEDGMENT

OM transliteration and integrated editor have been developed by a large number of people at the Multimedia Systems Lab at the Supercomputer Education and Research Centre, Indian Institute of Science and at the ISRI, Carnegie Mellon University. Of particular mention are the names of Sravan Kumar, Jiju Verghese, Sheik, Tina Joseph and Umi who contributed towards specific Indian languages. Jiju Verghese developed the Web interface and the Java standalone version was developed by Atul Kumar.

References

Balakrishnan, N., Reddy, R., Ganapathiraju, M., 2005. Digital Library of India: A testbed for Indian Language Research. IEEE Technical Committee on Digital Libraries Bulletin: Special Issue on Asian Digital Library Research (In Press).

Jayaraman, A., Sangani, S., Ganapathiraju, M., 2004. OmSE: Tamil Search Engine. Proc. Tamil Internet Conference, Singapore.

The statistics in the first 2 sections were collected from the following pages:

Bose, D.K., 2001. Rural India–Wired Up? View Point Online Magazine, 4. http://www.ogilvy.com/viewpoint/view.ko.php?id=16216&iMagaId=6.

Excerpts from the 1906 edition of Linguistic Survey of India (Telugu). http://www.engr.mun.ca/~adluri/telugu/language/usage/grierson/introduction.html.

Krishnan, G., 2002, Challengers in Rural Marketing, Strategic Marketing Forum. http://www.etstrategicmarketing.com/smJune-July2/forum.htm.

Smith, M., 2000. India's Chance to Lead the World. For a Change Magazine. http://www.forachange.co.uk/index.php?stoid=168.

Well, D.H., 2001, Milestones: A Road Map to the Indian Middle Class APF Reporter, **20**(1). http://www.Davidhwells.com/PhotoEssays/globalindia/middleclas/default.html.

SiliconIndia, 2005, $100 Computer Coming from India. http://www.siliconindia.com/shownewsdata.asp?newsno=28077&newscat=Technology.