International Journal of Advanced Research in IT and Engineering

ISSN: 2278-6244

DIFFERENT WRITING SYSTEM CHARACTER MAPPER

A LOW COST, OPEN SOURCE EDUCATIONAL APPROACH

Hardeep S. Jawanda*

Abstract: *Script-based languages from different writing systems demand information exchange to bridge gap in script based languages from different writing system. Here we present a light weight tool to accomplish task of information exchange for development of both the languages namely Gurmukhi and Shahmukhi. The technique presented in Regional Language character Mapper under Open Source Technologies (RLCMOS) is a significant low cost tool to map the information between the two script based languages. The technique used here is a blend of two other techniques, namely Corpora Based and Rule Based. This technique becomes a unique one as it generates its corpora from Gurmukhi Education Literature. RLCMOS is kept as an open source by using only open source tools and the developed tool itself is open source. Gurmukhi and Shahmukhi are identified forms of Punjabi script based languages coming from Syllabic / Abugidas and Abjad writing systems respectively. There is a considerable phonetic equivalence between both languages. RLCMOS will be useful in machine translation, cross-lingual information retrieval, multilingual text and speech processing between both languages.*

Keywords: *Gurmukhi, Punjabi, Shahmukhi, Translation, Corpora, Abugidas, Abjad.*

*Head of Computer Engineering Department, Guru Nanak Dev Polytechnic, Ludhiana

I. INTRODUCTION

Script-based regional languages from entirely different writing system with similar phonetics need information exchange. To explore and to enhance the available knowledge and information of regional languages , this is a considerable tool in sustaining the heritage and development of languages. It is seen that very less or no work on Gurmukhi-Shahmukhi, Punjabi language pair is available so far based upon literature survey & internal search [all search engines]. The users of both languages are able to understand the verbal expressions of each other but are unable to comprehend written Punjabi in respective Gurmukhi or Shahmukhi script. The presented tool "Regional Language character Mapper under Open Source Technologies" (RLCMOS) is a significant step towards mapping of the information from Gurmukhi script to Shahmukhi script. To allow the wide spread usage of language and its development , RLCMOS is kept as an open source.

II. WRITING SYSTEM

A writing system is a permanent symbols representation of voice or vocal sounds. Different sounds in different languages mean different things. The sounds when combined result in words and words lined up under rules form sentences. Some special symbols (vowels) are used to give stress / put nasal effect to the sound. There are some writing systems were same sounds / combination of sounds represent different symbols For eg. Hosla (The Courage) sound is same in both the writing systems , as these writing systems are for same language – Punjabi.

hOslf حوصلہ

Gurmukhi Shahmukhi

Writing systems can be divided into two main types: those that represent consonants and vowels (alphabets), and those which represent syllables (syllabaries), though some do both. There are a number of subdivisions of each type, and there are different classifications of writing systems in different sources. Consonantal Alphabet or Abjad Consonantal alphabets are also known as abjads, and are all descendents of the Proto-Sinaitic script. In a "pure" consonantal alphabet, vowels are not written. However, nearly consonantal alphabets use certain conventions to Syllabic Alphabet or Abugida. South Asian scripts such as Brahmi and its descendents fit into both syllabary and alphabet. It is syllabic because the basic sign contains a consonant and a vowel. However, every sign has the same vowel, such as /a/ in

Brahmi. To make syllables with a different vowel, you add special markings to the basic sign, which is somewhat like an alphabet. Hence the name "syllabic alphabet". Nearly all the sounds in a language can be represented by an appropriate consonant and vowel alphabet. [1]

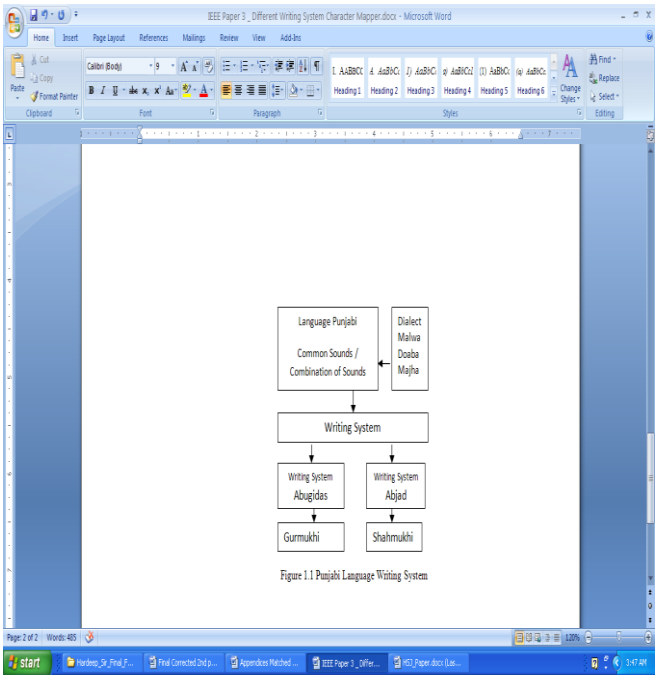

Figure Punjabi Language Writing System

III. CHARACTER MAPPING TECHNIQUE

A. Transliteration Phases

The transliteration system is virtually divided into two phases. The first phase performs pre-processing and the second phase performs post-processing.

Phase I (pre-processing)

- In preprocessing the input text is subjected to tokenizer which separates each word from another in the form of tokens.
- These tokens are then processed for any consecutive repeated occurrences.
- Tokens are also checked for mistyped or unwanted characters.
- Unwanted characters are deleted and Required characters are inserted.
- The input text made completely free from errors before transliteration.

Phase II (post-processing)

- Identify Proper Nouns (Names , Places) and replace them using name corpus.
- Replace char to char from Shahmukhi - Gurmukhi Dictionary.

- Replace word to word from Shahmukhi – Gurmukhi Dictionary.
- Bi-gram model is used after word to word replacement to find the suitability of the transliterated text.

- Replace common sentence to sentence from Shahmukhi – Gurmukhi Dictionary.

B. System Structure

The structure of the mapping technique is shown in figure 2, has the following stages through which the source text is passed.

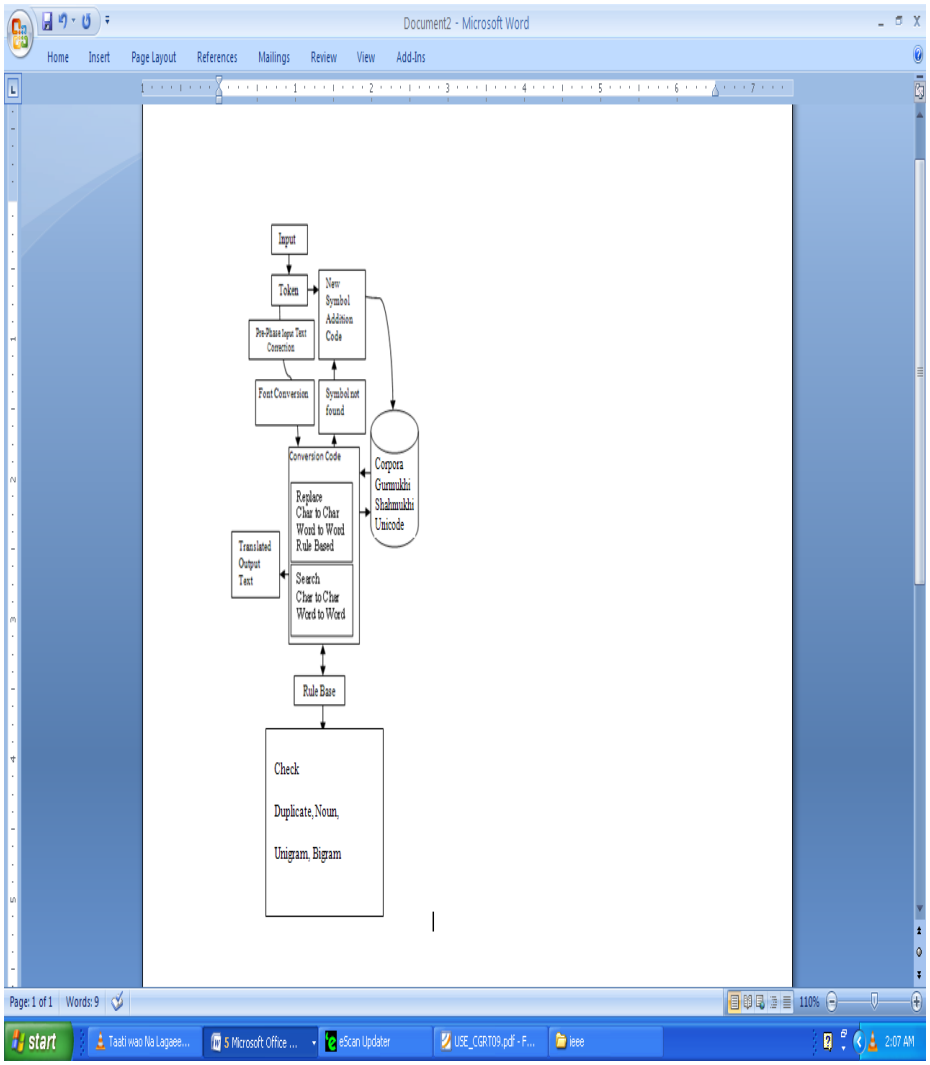

Figure Structure of Mapping Technique

C. Input Text Uniformity

Text normalization brings uniformity in various inputs that may be in number of different formats in the form of ASCII based fonts to represent Punjabi text. Each font varies in assigning ASCII code to Punjabi Alphabets. This is a cause of problem while searching a text in the corpora. Therefore, input text is first converted into Unicode format as a universal

standard. It gives us three fold advantages; first it will reduce the text scanning complexity. Secondly it also helps in internationalizing the system as if the output is in Unicode format then it can be used in various applications in various ways. Thirdly, it eases the transliteration task.

ASCII Based Punjabi Fonts	Latin Character A
Anmol Lipi	A → ਅ
DRChatrik	A → ਓ

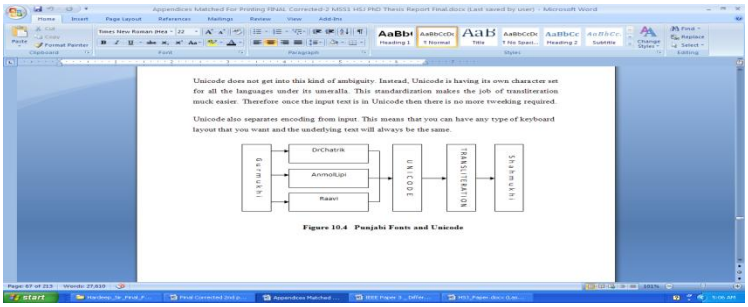

Figure Input Text Uniformity

D. Tokenization

Once the whole text is scanned, next step is to break-up the data into sentences. Individual words or tokens are extracted to find the equivalent in the target language. In other words mapping is done once the tokenization is complete.

E. System Specifications

To increase the usability of RLCMOS and improve its educational development, the whole tool it is proposed to keep it as open source. Further it is proposed to design it platform free and use Unicode code –set.

F. Open source

The code is accessible and is made as open source, so that further improvements can be contributed. Thus further development of RLCMOS is sited through free contributors. Till now also the open source path has been followed.

IV. CORPORA DEVELOPMENT

A large base of Punjabi Gurmukhi Corpora has been developed. This corpus has been developed in view with Unicode implementation and the education base. A Gurmukhi

document of above 6000 words is used to develop unigram and bigram education base corpora. The corpora is in six different forms , as given below.

- Gurmukhi – Shahmukhi - Unicode
- Gurmukhi – Shahmukhi – Nouns
- Gurmukhi – Shahmukhi – Words
- Gurmukhi – Shahmukhi – Sentence
- Gurmukhi – Unigram (Education Base)
- Gurmukhi- Bigram (Education Base)

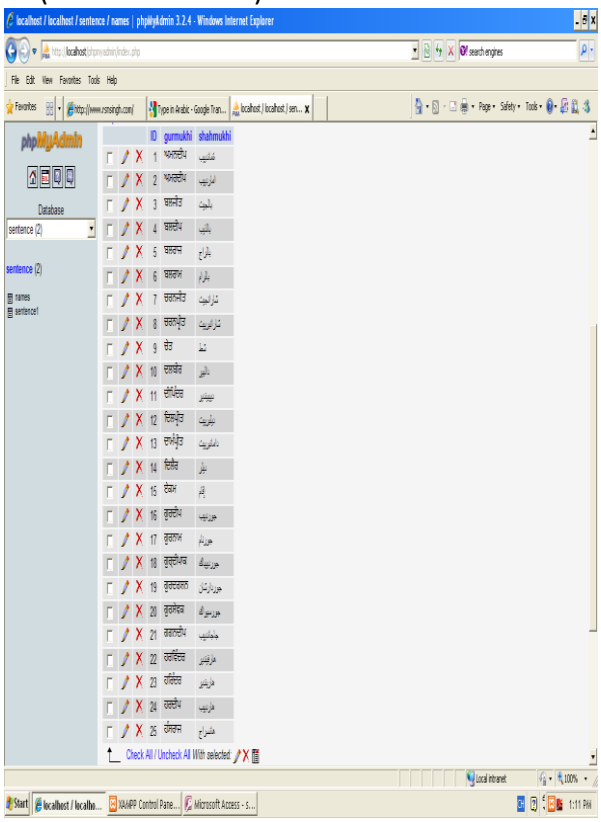

Figure Gurmukhi Shahmukhi Noun Corpus

S.No	Unigram	Occurrences
1	ਨਾਲ	72
2	ਬੱਚੇ	74
3	ਗਿਆ	20
4	ਨਹੀਂ	36
5	ਜਿਸ	18
6	ਸਿੰਘ	18
7	ਸਿੱਧੂ	2
8	ਪੰਜ	10
9	ਆਬਾਂ	1
10	ਦੀ	142

Table Part of Gurmukhi Unigram

Corpus Type	Number of Entries
Gurmukhi – Shahmukhi - Unicode	100
Gurmukhi – Shahmukhi – Nouns	100
Gurmukhi – Shahmukhi – Words	4927
Gurmukhi – Shahmukhi – Sentence	100
Gurmukhi – Unigram (Education Base 6556 word Gurmukhi Document)	500
Gurmukhi- Bigram (Education Base 6556 word Gurmukhi Document)	20

Table Corpora Base

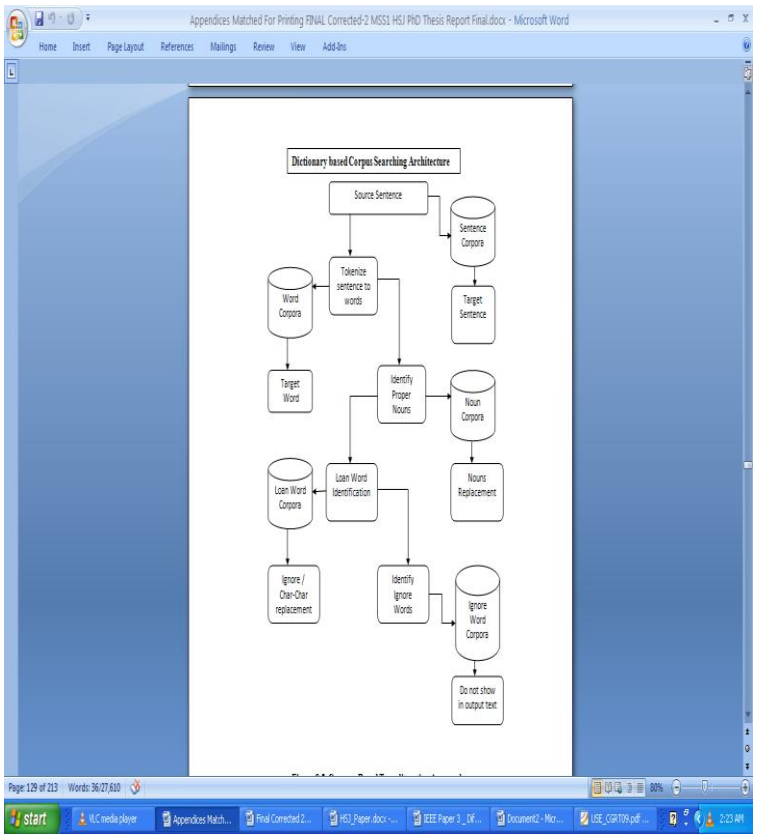

Figure Corpora Searching Architecture

V. RULE BASE DEVELOPMENT

Transliteration with corpus word matching is a straight forward. The rules are made very specific to Gurmukhi language. Rule structure can be implemented in variety of ways. Rules can be implemented through macro replacement or programming or manually also. Here

we have implemented the rules in software tool presented. The accuracy starts to degrade as and when there is ill structured input or corpora does not provide correct match. To cater to such kind of situations Rule are framed. These rules correct the source input wherever it possible. For example rules can be, do not replace latin number , as these are now universally accepted , replace font specific tokens with Unicode equivalent, adjust spaces in prefix and suffix. Once rules are applied on input text , output text can be checked and rules applied on it for minor corrections. Rule-based methods need rules to be manually framed for specific language. Thus it may not cover the entire possible combinations of tokens to be mapped / replaced [1] (Karimi, 2007).

VI. RELATED WORK

Most of the methods used for transliteration purpose are based on statistical approaches. As transliteration is not a new problem. The major techniques for transliteration can be broadly classified into two categories, viz. grapheme-based and phoneme-based approaches. [2]Knight et al. (1997) developed a phoneme-based, statistical model using finite state transducer that performed back-transliteration using transformation rules. [3] Abdul Jaleel and Larkey (2003) demonstrated a simple, statistical technique for building an English-Arabic transliteration model using Hidden Markov Model (HMM) and alignments obtained from GIZA++ [4](Och. and Ney, 2003). Some of the works under proper noun conversion is 'Named entity recognition' - Though Named Entity Recognition (NER) is a known research area [5](e.g. MUC-6 1995, Daille & Morin 2000), multilingual Named Entity Recognition is quite new.

In the context of Indian languages, [6] Aswani and Gaizauskas (2005) have used a transliteration similarity based technique to align English-Hindi parallel texts. They used character based direct correspondences between Hindi and English to produce possible transliterations. Then they apply edit distance based similarity to select the most probable transliteration in the English text. However, such methods can only be appropriate for aligning parallel texts as the number of possible candidates is quite small. This work is implicitly based on transliteration as alignment, but it is different from the alignment-based approach that we present in this article and it does not explore the idea beyond alignment of parallel corpus. [7]Malik (2006) proposed a rule-based method of transliterating Punjabi language words from Shahmukhi (Arabic script) to Gurumukhi script (derivation of Landa,

Shardha and Takri, some old scripts of the Indian subcontinent). [8]Ekbal et al. (2006) used a modified source-channel model for Bengali-English machine transliteration.

VII. IMPLEMENTATION

The implementation needs a strong corpus base. The transliteration system is virtually divided into two phases. The first phase the input text made completely free from errors before transliteration. In second phase sentence / word lookup and replacement is done from Shahmukhi - Gurmukhi corpus. Java is primarily used as the core tool development language. MySql is preferred for corpus storage. For the development purpose Eclipse and Netbeans are used. All the development tools are carefully chosen to make it simple, low cost and open source. All these tools have very good integration. This is because eclipse is especially used for java and MySql as a database get integrated very easily.

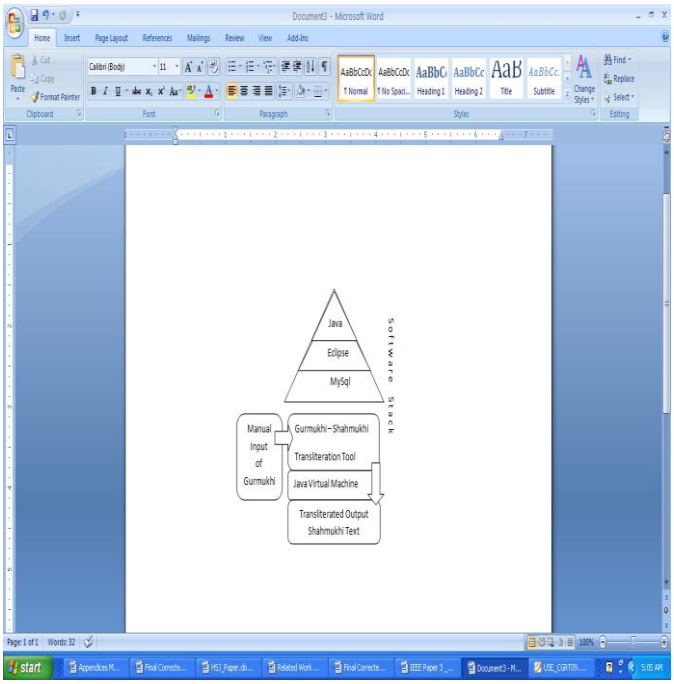

Figure Implementation Model

VIII. RESULTS

The results show that the efficiency and the accuracy are directly related to the corpus base. As the corpus increases (i.e. the entries at all levels of corpus are increased as mapping and replacing rate increases), the accuracy of the presented tool increases and the efficiency shows a slight dip as more number of entries are to be scanned. The major success rate of transliteration has been with nouns. This is because new names barely come up and to make a complete noun corpus becomes easy. The word and sentence mapping meet a

lesser success rate as sentence formation is a complex task and a corpora is difficult to make. The below table .. shows success rate results.

S.No.	Type	Transliteration Success rate
1	Sentence	5-10 %
2	word	10-20 %
3	Bigram	5 %
4	Noun	

Table Mapping Success Rate

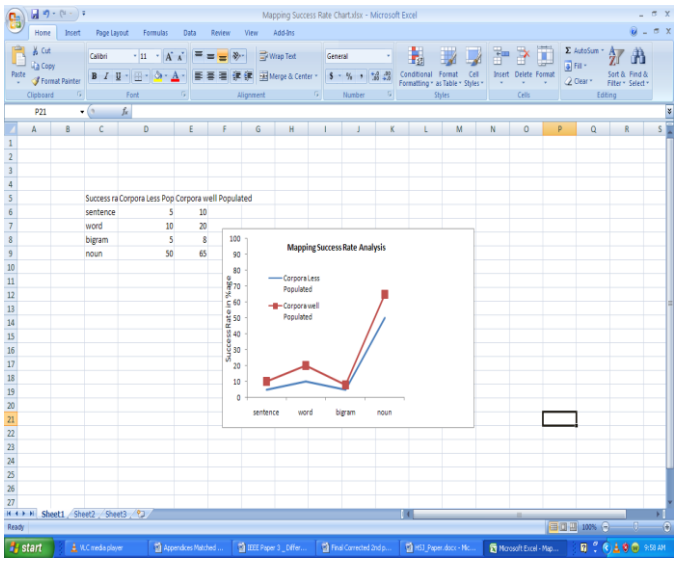

Figure Mapping Success Rate

IX. CONCLUSION

The software presented here is a lightweight tool and does the initial job of bridging gap between Gurmukhi and Shahmuki languages. A large corpora and well structured rules are the major tasks for making the mapping success rate high. The accuracy starts to degrade as and when there is ill structured input and corpora does not provide correct match.

ACKNOWLEDGMENT

Hardeep Singh Jawanda thanks Kirpal Singh Pannu for providing his analysis and database for use and testing of this work.

REFERENCES

[1] Sarvnaz Karimi, "Effective Transliteration," Ballarat, Australia, 2007.

[2] Knight K., Graehl J., “Machine Transliteration”, Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Morristown, NJ, USA, p. 128-135, 1997.

[3] Abdul Jaleel N., Larkey L. S.,2003 , “Statistical Transliteration for English-Arabic Cross Language Information Retrieval”, CIKM ’03: Proceedings of the Twelfth International Conference on Information and Knowledge Management, ACM, New York, NY, USA, p. 139-146,

[4] Och F. J., Ney H., “A Systematic Comparison of Various Statistical Alignment Models”, Computational Linguistics, vol. 29, n° 1, p. 19-51, 2003.

[5] Daille Béatrice, Morin Emmanuel, (2000) Reconnaissance automatique des noms propres de la langue écrite: les récentes réalisations, Traitement Automatique des Langues (TAL), Vol. 41(3), pp.601-622, 2000.

[6] Aswani N., Gaizauskas R., “A Hybrid Approach to Align Sentences andWords in English-Hindi Parallel Corpora”, Proceedings of the ACL Workshop on Building and Using Parallel Texts, Association for Computational Linguistics, Ann Arbor, Michigan, p. 57-64, June, 2005.

[7] Malik, M. A. (2006, July). Punjabi Machine Transliteration. 21st International Conference on Computational Lingustics (pp. 1137-1144). Sydney: ACL.

[8] Ekbal A., Naskar S. K., Bandyopadhyay S., “A Modified Joint Source-Channel Model for Transliteration”, Proceedings of the COLING/ACL on Main Conference Poster Sessions,

[9] Kaur, D. H. (2008, June 18). *ਮਾਂ-ਬੋਲੀ - ਇਕ ਡਾਕਟਰੀ ਦ੍ਰਿਸ਼ਟੀਕੋਣ.* Retrieved July 15, 2011, from www.wichaar.com : http://www.wichaar.com/news/258/ARTICLE/6261/2008-06-18.html

[10] Lo., L. K. (1996). *Types of Writing Systems*. Retrieved July 25, 2011, from http://www.ancientscripts.com : http://www.ancientscripts.com/ws_types.html