Machine Translation of Bi-lingual Hindi-English (Hinglish) Text

R. Mahesh K. Sinha
Indian Institute of Technology, Kanpur
rmk@iitk.ac.in

Anil Thakur
Indian Institute of Technology, Kanpur
anilt@cse.iitk.ac.in

Abstract

In the present communication-based society, no natural language seems to have been left untouched by the trends of code-mixing. For different communicative purposes, a language uses linguistic codes from other languages. This gives rise to a mixed language which is neither totally the host language nor the foreign language. The mixed language poses a new challenge to the problem of machine translation. It is necessary to identify the "foreign" elements in the source language and process them accordingly. The foreign elements may not appear in their original form and may get morphologically transformed as per the host language. Further, in a complex sentence, a clause/utterance may be in the host language while another clause/utterance may be in the foreign language.

Code-mixing of Hindi and English where Hindi is the host language, is a common phenomenon in day-to-day language usage in Indian metropolis. The scenario is so common that people have started considering this a different variety altogether and calling it by the name Hinglish. In this paper, we present a mechanism for machine translation of Hinglish to pure (standard) Hindi and pure English forms.

1 Introduction

Code-mixing is a frequently encountered phenomenon in day-to-day natural language communication in metropolises specially among educated people. The phenomenon is so common that this is often considered a different (emerging) variety of the language. Two terms are generally used to describe this phenomenon. The term "code-mixing" is used to describe mixing of elements from different languages within a sentence whereas the term "code-switching" describes mixing of elements from different languages at the clause level in a discourse. According to Bhatia and Ritchie (1996) "code-mixing refers to the mixing of various linguistic units (morphemes, words, modifiers, phrases, clauses and sentences) primarily from two participating grammatical systems within a sentence". However, these terms have been used interchangeably in the relevant literature (Bhatt 1997). It is also important to note that in many a case, it is difficult to decide whether it is a case of borrowing or code-mixing (Singh 1985, Poplack 2001). This makes the attempts to understand the structure of code-mixing unwieldy and descriptively inadequate (Poplack 2001). In this paper, we have considered the code mixing of Hindi and English.

The English words or phrases that occur in the Hindi-English code-mixed texts can be of different grammatical categories. Some sample examples[1] are given in (1) depicting the extent of mixing. In (1a), 'boiled' is an adjective, 'rice' a noun and 'cook' occurs as a verb followed by a Hindi auxiliary verb *kiyaa* ('did'). In (1b), 'report*oN*' is plural form of English noun 'report' as per Hindi morphology followed by Hindi postposition '*ko*', 'carefully' is an adverb from English, and 'prepare' is an English verb followed by a Hindi auxiliary verb *kiyaa* ('did'). In (1c), a whole clause ('the *Sadhu*s did not come') from English occurs along with a Hindi clause ('*usane bahut* wait *kiyaa'*) in a compound sentence with English conjunction 'but'. The word '*Sadhu*s' in the English clause is plural form of Hindi noun '*Sadhu*' as per English morphology.

(1) a. *usane* boiled rice cook *kiyaa*.
he-ERG boiled rice cook did
'He cooked the boiled rice.'

b. *usane* report*oN ko bahut* carefully prepare *kiyaa*.
He-EEG reports to very carefully prepare did.
'He prepared the reports very carefully.'

c. *usane bahut* wait *kiyaa*, but the *Sadhu*s did not come.
He-ERG very wait did, but the Sadhus did not come.
'He waited for long, but the *Sadhu*s did not come.'

It is evident from these examples that code mixing in Hindi-English is quite varied. Such a mixing is very common and is particularly prevalent among the educated section of the

[1] The Hindi words have been shown in *italics*.

society and in metropolises. The scenario is so common that people have started considering this a different variety altogether and calling it by the name Hinglish [2]. In this paper, we present a mechanism for machine translation of Hinglish to pure (standard) Hindi and pure English forms. The sentences of the type (1c) are most general and have been referred as CCM (complex code-mixed) in the text. The sentences of the type (1a) and (1b) are referred as SCMH (simple code-mixed Hindi) whereas the English clause of example (1c) is referred as SCME (simple code-mixed English). It may be noted that SCMH will have a Hindi main verb whereas in case of SCME, it will be an English main verb. In section 2, we present some of the constraints on code-mixing across language. Sections 3 and 4 present the major system design considerations and the translation schema respectively.

2 Mixing Constraints

The phenomenon of code-mixing has been extensively studied in the literature, both from the sociolinguistic and grammar perspectives. In the literature, different patterns of code-mixing have been pointed out. Scholars have also examined the different constraints on the type of elements that can be code-mixed (Kachru 1978, Pfaff 1979, Sankoff and Poplack I981, Singh 1985, Bhatia and Ritchie 1996). Bhatia and Ritchie (1996) discuss the issue of code-mixing across languages and examine the different types of constraints on code-mixing that have been proposed by different scholars in the literature. Kachru (1978) has also mentioned some constraints. Several of these constraints have been subsumed in broader constraints in later literature (Singh 1985, Di Sciullo, Muysken, and Singh, 1986, Belazi, Rubin, and Toribio 1994). Many of these constraints, in later literature, have been shown to have significant amount of counter examples. Agnihotri (1998) discusses a number of examples of Hindi-English code-mixing and shows that a number of constraints that have been proposed in literature on code-mixing are violated in the case of Hindi-English. This shows that the phenomenon of code-mixing needs further exploration for determining constraints. However, the existing constraints certainly present a guideline for the purpose of the identification of the nature of the "foreign" elements in a language. Without going into details of these constraints and their theoretical motivations, we discuss some of them here with a view to examining their usefulness in the context of issues discussed in this paper.

i. The Equivalence Constraint: The equivalence constraint (Pfaff 1979, Poplack 1981, Sankoff and Poplack 1981, Muysken 2000, Poplack 2001) states that the switches will not violate a syntactic rule of either language, that is, at the points at which the surface structures of the two languages map onto one another. In Hindi, where the word order in the sentence is different (SOV) from that of English (SVO), this constraint can be shown to be violated very commonly. A number of counter examples have been presented to show that the equivalence constraint does not hold between languages of diverse typological characteristics (Bentahila and Davies 1983). It has been shown that equivalence constraints, along with some other constraints can be subsumed in the general constraint of government (Singh 1985, Belazi, Rubin, and Toribio 1994).

ii. The Free Morpheme Constraint: The free morpheme constraint (Poplack 1980, Sankoff and Poplack 1981) prohibits a switch between a stem and a bound morpheme from different languages. In other words, this constraint does not allow the words from language L2 to inflect according to the grammar rules of language L1 and vice-versa. However, as pointed out by Agnihotri (1998) this constraint is very frequently violated in Hindi-English code-mixing. However, there is a question as to whether such elements that inflect according to the grammar rules of the host language can be categorized as lexical borrowings. All the borrowed elements do not necessarily inflect according to the rules of the host language. It is difficult to categorize what kind of elements can inflect according to the grammar rules of L1 and what cannot. For instance, *TikaTeN* and 'tickets' both are possible. There are a large number of English words (mostly nominal) that are used in Hindi sentences according to the grammar rules of the Hindi language.

Even if a number of such English words can be categorized as lexical borrowing which (ideally) can be entered in the Hindi lexicon, it is not possible to enter/cover all such words in the Hindi lexicon and hence these cases have been dealt with as examples of code-mixing in this paper.

iii. The Closed Class Constraint: The closed class constraints (Sridhar and Sridhar 1980, Joshi 1985) states that the elements categorized as closed class of grammar such as possessives, ordinals, determiners, pronouns and other limiting adjectives are not allowed in code-mixing. Similar constraints like determiner constraint and conjunction constraint (Kachru 1978) can be subsumed in this constraint. The determiner constraint states that the

[2] Although Hinglish is very commonly used in Indian metropolises, it is fully understood by less than 25% of the total Hindi speaking population.

determiner and the noun in a noun phrase cannot be from different languages.

The conjunction constraint states that a conjunction of L1 cannot be used to conjoin words, phrase or clauses of L2 and the vice-versa. Agnihotri (1998) does present a counter example that appears to be quite uncommon in Hindi-English code-mixed text.

Thus we can safely say that this constraint holds in Hindi-English code-mixed text. For instance, prepositions, determiners, possessives, pronouns, quantifiers, etc. generally do not occur as code-mixed elements in Hindi-English code-mixed text. Hence, the constructs marked with '*' in example in (2) are not quite possible.

(2) *He *ne**my *kitaab* *the *mej* *on *rakhii*.
He-ERG my book the table on kept.
'He kept my book on the table.'

Thus if the head noun is of Hindi, pronouns and the pre-nominal modifiers such as determiner, possessives, and numerals cannot be from English.

iv. The Dual Structure Principle: The dual structure constraint (Sridhar and Sridhar 1980) states that the internal structure of L2 constituent need not conform to the constituent structure rules of L1 (host language) if its placement obeys the rules of the host language. Thus a Hindi example as in (3) is possible for code-mixing.

(3) *mere vichar meN* his visiting her *Thiik nahiiN hE*.
my opinion in his visiting her proper not be.PR
'In my opinion, his visiting her is not proper.'

The above discussion on constraints shows that code-mixing is not completely arbitrary. We have used these constraints in devising mechanism for recognizing constituents and for translating Hinglish to Hindi/English.

3 Major System Design Considerations

It is interesting to note that the complexity of the system for processing Hinglish text depends largely upon the script used for writing the Hindi and English parts of the text. In this context, following scenarios emerge:

i. Devanagari (script for Hindi) is used to write both the Hindi and English parts in code mixed text. In such a case, the English words have to be transliterated into Devanagari. As English words are not written the same way as spoken, this process of transliteration introduces errors. Thus the transliterated English word may not be found in a standard English dictionary. It may be noted that the unknown words (unknown both in Hindi and English) becomes difficult to determine. We need to generate aliases of all such unknown words based on modelling of transliteration error to find possible mapping to a valid English word.

ii. Devanagari is used to write Hindi and Roman is used for writing the English part of the code mixed text. In this case, we have no problem in identifying Hindi, English and unknown words.

iii. Roman is used to write both Hindi and English parts in the code mixed text. This scenario is similar to the one outlined in (i) above. However, as Hindi is written the same way as spoken, it is now a lot easier to deal with the transliteration errors.

In this work, we have considered the mixed text written in Roman. The spellings of English words are as used in Standard English and Hindi words are transliterated into Roman with appropriate phonetic encoding as expected in the Hindi lexical database. When the mixed element is a word or a phrase, we consider this type of mixing as 'simple code-mixing' (SCM) whereas in 'complex code-mixing' (CCM), the mixed element is a clause.

4 Translation Schema

The translation schema consists of the following seven steps:

i. Each word of the input complex code-mixed (CCM) Hinglish sentence is fed to Hindi morphological analyzer. The morphological analyzer yields part of speech information for each of the words and marks the words that are unknown in Hindi. It should be noted that it is quite possible that a Hinglish word gets labeled as a Hindi word even though it may actually be an English word. This may happen due to Roman coding used for inputting Hindi.

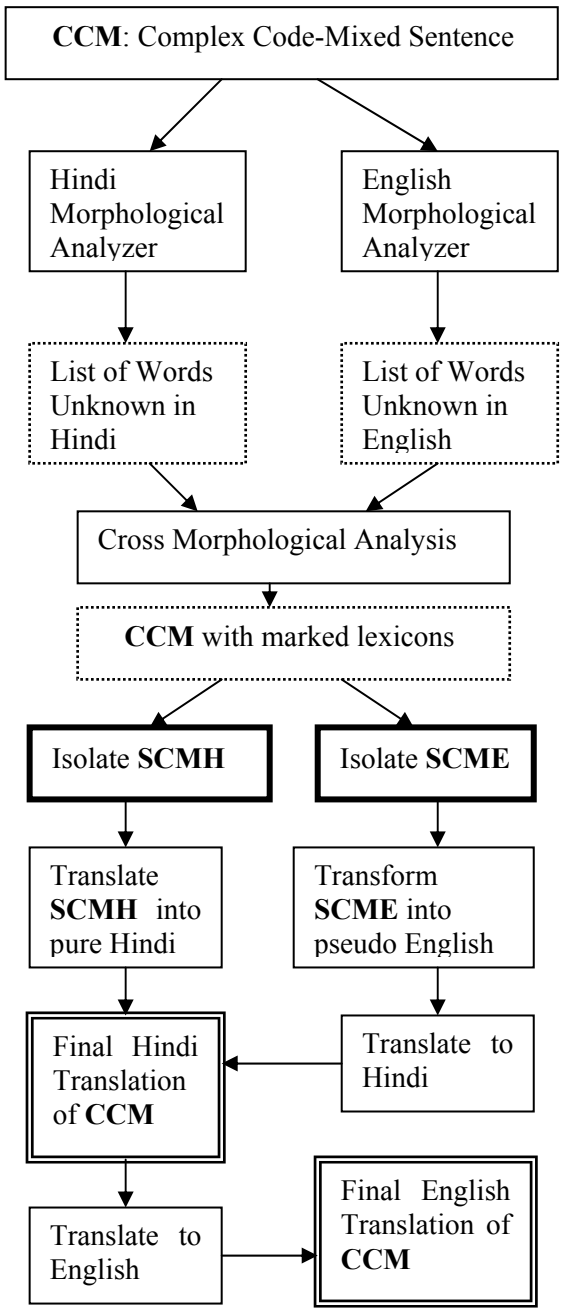
ii. Each word of the input complex code-mixed (CCM) Hinglish sentence is fed to English morphological analyzer. The morphological analyzer yields part of speech information for each word and marks the words that are unknown in English. It should be noted that it is quite possible that a Hinglish word gets labeled as an English word even though it may actually be a Hindi word. This may happen due to Roman coding used for inputting Hindi.

iii. The words that remain unknown (Sinha 2001) in steps (i) and (ii) above, are marked for cross morphological analysis for plural noun forms. Each unknown word is analyzed for being possible plural form of a Hindi word as per English morphology and vice-versa. The words that remain unknown even at this stage are marked as proper names. It may be noted that the steps (i) & (ii) may result in labeling a lexical item both as a Hindi as well as an English word.

iv. The input complex code-mixed (CCM) Hinglish sentence is segmented into simple code-mixed Hindi (SCMH) and simple code-mixed English (SCME) parts.

v. Translate SCMH into pure Hindi language.

vi. Transform SCME into pseudo English form (see section 4.2.2) and then translate to pure Hindi language [3].



SCMH: Simple Code-Mixed Hindi
SCME: Simple Code-Mixed English

Figure 1: The System Flow-diagram

vii. If the target language is pure Hindi, it is obtained by simply re-composing the translations as obtained above. For pure English, translation is obtained by translating the pure Hindi text.

The system flow is depicted in figure 1.

It is interesting to note that the phrase order of the translation more or less remains the same as that of the original code mixed text. Only the adverbial phrase moves before the noun phrase in case of translation to pure English (see the translation of illustration-III in sections 4.1 and 4.3).

Given below is an illustration of the above steps:

Input CCM Hinglish sentence:
When I reached *Baarataghara*, sweepers room*oN ko* clean *kar rahe the aur baraati*s were taking snacks in *pandaal*.

Step-(i): Hindi words identified and labeled: *ko*: post-position; *kar*: verb-aux; *rahe*: verb-aux, verb-aux; *the*: verb-aux; *aur*: conjunction; *pandaal*: noun.

Step-(ii): English words identified and labeled: when: conjunction; I: pronoun; reached: verb (root: reach); sweepers: noun (plural); clean: verb (root: clean); the: art; were: verb-aux; taking: verb (root: take); snacks: noun (plural); in: preposition.

Step-(iii): The unknown words at this stage are: '*Baarataghara*', 'room*oN*', '*baraatis*'. '*Baarataghara*' is marked as a proper noun after cross morphological analysis. 'room*oN*' is identified as pluralized form of English word 'room' by Hindi morphological analyzer. One of the ways to derive plural form of a Hindi noun is to suffix it with '*oN'*. '*baraatis*' gets identified as pluralized form of Hindi word '*baraati*' by English morphological analyzer.

Step-(iv): The input bilingual Hinglish sentence gets segmented into:
Simple Code-mixed Hindi (SCMH):
sweepers room*oN ko* clean *kar rahe the*
Simple Code-mixed English (SCME) parts:
(a) When I reached *Baarataghara*
(b) *baraatis* were taking snacks in *pandaal*

Step-(v): Translate SCMH into target language:
Target Hindi: *saphaaikarmii kamaroN ko saaph kar rahe the*.
Target English: sweepers were cleaning the rooms.

Step-(vi): Transform SCME into pseudo English:
(a) When I reached *Baarataghara*
(b) np1 were taking snacks in np2.
Pure Hindi translation:
(a) *jab mEN Baarataghara pahuNchaa*.
(b) *baraati pandaal meN alpaahaar le rahe the*.

Step-(vii): Compose the target language:
Target Hindi: *jab mEN Baarataghara pahuNchaa, saphaaikarmii kamaroN ko saaph kar rahe the aur baraati pandaal men alpaahaar le rahe the*.
Target English: When I reached Baarataghara, sweepers were cleaning the rooms and the

[3] The primary reason for translating first to pure Hindi is motivated by availability of Hindi to English translation system with us. Besides this, Hinglish is Hindi centric. Alternatively, SCME could be translated to pure English and merged with SCMH translated part.

participants of the marriage procession were taking snacks in tent.

4.1 Isolating SCMH and SCME from CCM

Although, it is believed that there exists a definite grammar for switching between SCMH and SCME, its formulation is not very straightforward. Our strategy for extracting these constituents is based on certain well tested heuristics and shallow parsing. It is observed that the switching is usually marked with a silence (a comma) or a conjunction of one of the languages. We exploit the property of Hindi being a verb ending language in identifying the switching boundaries. Figure 2 depicts the basic strategy.

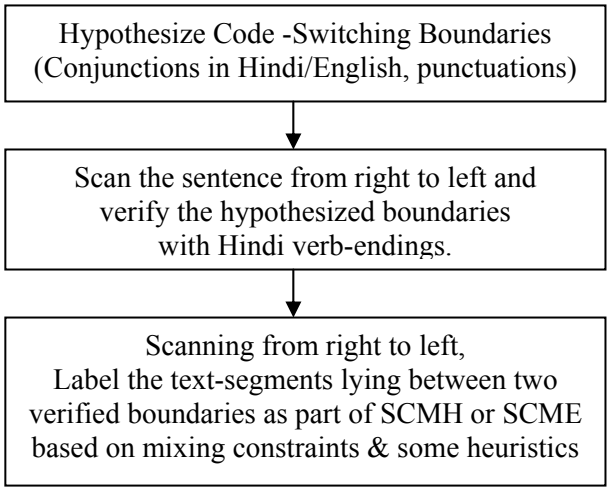


Figure 2: Isolation of SCMH and SCME

The steps are as outlined below:

i. Hypothesize all conjunctions of Hindi and English as plausible boundaries. All commas excepting those following conjunctions are also considered plausible switching boundaries. These positions are stored in an array i(n) from right to left where n is the total number of hypothesized switching boundaries. The position of front end of the sentence is stored in i(n+1).

ii. If n=0 then there is no code switching. In this case, the entire sentence is taken as SCMH if it ends with a Hindi verb form, else it is taken as SCME. It may be noted that here we are not considering partial sentences.

iii. Start scanning the mixed sentence from the tail end side (right to left) and search for Hindi main verb-form after each $i(n)^{th}$ hypothesized switching boundary. Hindi verb infinitive or gerund forms are not considered as main verb forms. Store the position of tail of each Hindi main verb-form found in another array j(m) where m is the estimated number of SCMH in the sentence. If (m=0) then the entire input sentence is taken as SCME. Set j(m+1)= i(n+1).

iv. The text lying between $j(1)^{th}$ position to next hypothesized boundary to the right, is taken as initial SCMH. If m>0 and j(1) > 1, then the text lying between $j(1)^{th}$ position till end, is taken as SCME.

v. For k= 2 to m+1 do

Examine the text between the j(k)th to j(k-1)th positions:

(a) If the enclosed text does not contain an English main verb-form, the text under consideration is taken as part of the previous SCMH.

(b) Else, if there is an English main verb-form and there are no more hypothesized boundaries in the text under consideration, the text is taken as part of a SCME.

(c) Else, scan the enclosed text from right to left, move the text segment till the position of next hypothesized boundary to become part of the previous SCMH, if any one of the following conditions is met:

1. contains any post-position or inflectional marker of Hindi,
2. all the words belong to Hindi only,
3. the conjunction at the boundary is in Hindi,
4. an adverbial phrase in Hindi or English with coordinate conjunction at the boundary.

Given below are a few illustrations of the above mentioned procedure with different CCM Hinglish sentences. Here the conjunctions have been shown enclosed in curly parentheses and the Hindi sentence boundary by a right bracket symbol.

Illustration-1:

CCM: {When} I reached *Baarataghara* {,} sweepers room*oN ko* clean *kar rahe the*] {*aur*} *baraati*s were taking snacks in *pandaal*.

SCME: *baraati*s were taking snacks in *pandaal* (step iii)

SCMH: sweepers room*oN ko* clean *kar rahe the* (step v(c))

SCME: When I reached *Baarataghara* (step v(b))

Illustration-2:

CCM: The *pravachan* was going on since morning, {and} in the end {,} chief priest *ne maTh ke sabhii* devotees *ko* bless *kiyaa*,] {*aur*} finally {,} program end *huaa*.]

SCMH: finally program end *huaa* (step v(a))

SCMH: in the end chief priest *ne maTh ke sabhii* devotees *ko bless kiyaa* (step v(c4))

SCME: The *pravachan* was going on since morning (step v(b))

Illustration-3: CCM: We should not believe {*ki*} *is taaviij ko* {*yaa*} *us* ring *ko* wear *karane se* {,} we will get success.

SCME: We should not believe *ki is taaviij ko yaa us* ring *ko* wear *karane se*, we will get success (step ii)

Illustration-4:
CCM: *use* fancy clothes {*aur*} rich food *pasand hai*.]
It may be noted here that there are two possible segmentations at the surface level. Here the word '*use*' is an English verb/noun and it also represents a Hindi pronoun 'he'. This ambiguity is caused by the Roman coding used for Devanagari. Resolution of this ambiguity requires semantic analysis.
(a) taking 'use' as an English verb:
SCMH: rich food *pasand hai* (by step ii)
SCME: use fancy clothes (by step v(b))
(b) taking '*use*' as Hindi pronoun:
SCMH: *use* fancy clothes *aur* rich food *pasand hai* (by step v(a))

4.2 Translation to pure Hindi

4.2.1 Translating mixed Hindi (SCMH) part:

As pointed out earlier, SCMH may contain English noun phrase and adverbial phrase embedded in a Hindi sentence frame in addition to verbalization of English noun and verb. For translation to pure Hindi, all of these components have to be translated. It consists of the following steps:

i. Recognize English verb forms used in SCMH. For this, all English verbs that are followed by any derivative form of Hindi auxiliary verbs such as *kar* or *ho* are substituted with corresponding Hindi meaning. An English word that is both a verb and another part of speech, and is followed by any derivative form of Hindi auxiliary verbs such as *kar* or *ho,* is treated only as a verb. The tense, aspect and modality (TAM) of the Hindi auxiliary verb and a set of rules are used to derive corresponding pure Hindi form.

ii. Recognize conjunctions, noun and adverbial phrases of English from the substituted sentence as obtained in (i) above. We use finite state machine (FSM) formalism for this. FSM is designed to accept the permitted mixing while applying mixing constraints in individual constructs.

iii. Translate the recognized noun and adverbial phrases into pure Hindi using applicable transfer rules. Hindi translations are substituted in SCMH at their original place.

Given below are a few illustrations of the process:

Illustration-5: SCMH: sweepers room*oN ko* clean *kar rahe the*.

The English word 'clean' is a verb and is followed by Hindi auxiliary verb *kar*. The TAM of the verb part (clean *kar rahe the*) is past-continuous, active, normal mode. The Hindi meaning of English verb 'clean' is '*saaph kar*' and this gets transformed to '*saaph kar rahe the*' as per the TAM information. This yields the transformed sentence after step (i) as: sweepers room*oN ko saaph kar rahe the*. Now, there are two English/mixed noun phrases in this sentence: 'sweepers' and 'room*oN ko*'. The phrase 'sweepers' gets translated to '*saphaaikarmii*' (plural form of Hindi meaning of the English root word 'sweeper'). The second noun phrase is 'room*oN ko*' where the head noun is a Hindi plural form of English noun 'room'. This is translated as *kamaroN ko* (plural form of Hindi meaning of the English noun 'room').

The final translation obtained is as follows: *saphaaikarmii kamaroN ko saaph kar rahe the*.

Illustration-6: SCMH: ticket book *huaa*.
The English word 'book' is a verb (besides being a noun) and is followed by Hindi auxiliary verb *huaa* (past form of its root *ho*). The Hindi meaning of the verb 'book' is '*aarashit kar*'. Now one of the rules for transformation is that if an auxiliary '*kar*' is followed by '*ho*', then the auxiliary is taken as '*ho*' {verb + *kar* + *ho* => verb + *ho*}. Thus the verb part gets translated to '*aarashit huaa*' as per the TAM information.
The final Hindi translation obtained: TikaT *aarashit huaa*.

Illustration-7: SCMH: *vah* dance *karatii hai.*
The English word 'dance' is a verb (besides being a noun) and is followed by Hindi auxiliary verb *karatii* (inflected form of its root *kar*). The Hindi meaning of the verb 'dance' is '*naach*'. Accordingly, '*naach*' gets transformed as per TAM of '*karatii hai*' leading to pure Hindi form as '*naachatii hai*'. Noun meaning of 'dance' is ignored.

The final Hindi translation obtained: *vah naachatii hai.*

Illustration-8: SCMH: *usnane* bless *kiyaa.*
The English word 'bless' is a verb and is followed by Hindi auxiliary verb *kiyaa* (past form of its root *kar*). The Hindi meaning of the verb 'bless' is '*aashirwad de*'. Accordingly, '*aashirwad de*' gets transformed as per TAM of '*kiyaa*' leading to pure Hindi form as '*aashirwad diyaa*'.
The final Hindi translation obtained: *usnane aashirwad diyaa.*

It can be seen from the above that the strategy yields a satisfactory derivation of pure Hindi form of English verbs in case of verbs having single meaning. However, in case of polysemous verbs there is no mechanism provided for disambiguation. For disambiguation, we need to perform a more detailed syntactic analysis and consider verb semantics. For example, the mixed Hindi sentence as in illustration-9 will give both the transitive and intransitive meanings of the English verb 'boil'. This can be easily resolved with complete syntactic analysis of the sentence.

Illustration-9: SCMH: *usane duudh* boil *kiyaa*.

Translation to pure Hindi form:

a. *usane duudh ubaalaa.* {taking transitive meaning of 'boil' which is '*ubaal*'=>'*ubaalaa*' as per TAM of '*kiyaa*'}.

b. * *usane duudh ubalaa.* {taking intransitive meaning of 'boil' which is '*ubal*'=>'*ubalaa*' as per TAM of '*kiyaa*'}.

4.2.2 Translating mixed English (SCME) part:

As the matrix language for this part is English, it can only have a mixture of Hindi noun phrases and adverbial phrases besides conjunctions. These mixed phrases after being isolated, are substituted with dummy variables. These dummy variables carry the same syntactic category as that of the parent phrases. The transformed SCME sentence (referred to as pseudo English) is translated using conventional English to Hindi translation system. The chunk for each dummy variable is separately translated and finally substituted in the original sentence.

While translating the chunks corresponding to the dummy variables, the English nouns are simply substituted with their corresponding Hindi meaning. These phrases can also have an English verb in gerund or participle form. These are separately recognized and substituted with corresponding Hindi form. If a Hindi noun phrase is followed by an adverbial phrase, they are joined and labeled as a noun phrase. We use finite state machine (FSM) formalism for this.

Given below is an illustration of the process:

Illustration-10: SCME: We should not believe *ki is taaviij ko yaa us* ring *ko* wear *karane se*, we will get success.

Hindi conjunction: *ki* (substituted with dummy variable 'conj1').

Hindi noun phrase: *is taaviij ko yaa us* ring *ko* (substituted with dummy variable 'np1')

Hindi adverbial phrase: wear *karane se* (substituted with dummy variable 'adv1')

After translation of these components: conj1= *ki*; np1= *is taaviij ko yaa us anguuthii ko*; adv1= *pahanane se*; np2=np1 adv1.

Transformed SCME in pseudo-English: We should not believe conj1 np2, we will get success.

Hindi translation: *hameN nahiiN vishvaas karanaa chaahiye* conj1 np2, *hameN saphalataa milegii.*

Final Hindi translation: *hameN nahiiN vishvaas karanaa chaahiye ki is taaviij ko yaa us anguuThii ko pahanane se, hameN saphalataa milegi.*

4.3 Translation to pure English

The pure Hindi translations of both SCMH and SCME are re-composed as explained in the preceding section. This pure Hindi translation is then translated to pure English using a conventional Hindi to English translator.

5 Experimentation

The Hinglish translation system has been implemented by incorporating additional layer to the existing English to Hindi translation (AnglaBharti-II) and Hindi to English translation (AnuBharti-II) systems developed by Sinha (2004). It directly utilizes the Hindi and English lexical data-bases and morphological analyzers. The cross-lingual morphological analysis is implemented utilizing the already existing stemming functions incorporated into Hindi and English morphological analysis. The module for isolating SCMH and SCME from CCM utilizes the pre-existing modules of the two translation systems. The Hindi verb endings, their groupings and other syntactic units get evaluated the same way as in the Hindi to English translation system. An additional module incorporates interpretation of verbalization of English nouns/verbs with Hindi auxiliaries.

One day *mEN apanii* friend *se* dynastic politics *par baat kar rahii thii*. She said *ki* Rahul *aur* Priyanka *ko* politics *meN nahiiN aanaa chaahiye*. *mENne kahaa ki in logoN ko aanaa chaahiye* because they come from this environment. *unake* blood *meN* politics *hE*. Indira Gandhi *ek kushal* politician *aur* diplomat *bhii thii*. *vah* problems *ko* solve *karanaa iaanatii thii.*

a. Hinglish (mixed Hindi-English) input

ek din mEN apanii dost se vansgat raajaniiti par baat kar rahii thii. usane kahaa ki raahul aur priyankaa ko raajaniiti meN nahiiN aanaa chaahiye. mENne kahaa ki in logoN ko aanaa chaahiye kyoNki ve is parivesh se aate hEN. unake khuun meN raajaniiti hE. indiraa gaandhii ek kushal raajaniitigya aur kuuTaniitigya bhii thii. vah samasayaayoN ko hal karanaa jaanatii thii.

b. Pure Hindi translation (Romaized form)

One day I was talking with my friend on dynastic politics. He/she said that Rahul and Priyanka should not come in politics. I said that these people should come because they come from this environment. Politics is in their blood. Indira Gandhi was one skilled politician and diplomat also. She knew how to solve problems.

c. Standard English translation

Figure 3: A Sample Input and Output of Hinglish translation system

We have tested our system with inputs of mixed text transcriptions of narrations of arbitrary events from four independent speakers. A sample input/output of the system is given in figure 3. We observe that the strategy outlined here yields satisfactory acceptable results in more than 90% of the cases. Only in case of polysemous verbs, due to a very shallow grammatical analysis used in the process, the system is unable to resolve their meaning. These are generated as alternative forms for post-editing. With a detailed grammatical analysis, it is possible to resolve the meaning and/or automate the process of post-editing.

6 Conclusions

In this paper, we have examined the Hindi-English code-mixed text with a view to obtain machine translation for the mixed texts. We have reviewed different constraints on code-mixing that have been proposed in the literature on code-mixing/-switching. However, we have not gone into detail into the theoretical implications of the different constraints. We have examined them from the point of view of their usefulness in identifying/detecting the categorial status of the English words in predominantly Hindi texts. On the basis of the discussion of the patterns of occurrence of the different English words in the Hindi texts, we have devised strategies for their identification and translation to pure Hindi form and to pure English form. As mixing of Hindi and English usually takes place in verbal communication and in on-line chatting on the internet, such a mixed corpora is not available for elaborate testing. The testing has been performed by recording transliterated versions of narrations from a few subjects using existing components of Hindi to English and English to Hindi translation systems.

The strategy described here is equally applicable to all Indian languages as these are verb ending languages and have similar mixture of lexicons as in case of Hindi.

References

Agnihotri, R. K. 1998. *Mixed Codes and their Acceptability*. In Agnihotri, R. K. and Khanna, A. L. eds. Social Psychological Perspectives on Second Language Learning, 215-230, New Delhi: Sage Publication.

Belazi, M., Rubin, Edward J., and Toribio, Almeida J. 1994. Code Switching and X-bar Theory: The Functional Head Constraint. *Linguistic Inquiry* 25 (2):221-237.

Bhatia, Tej and Ritchie, William 1996. *Bilingual Language Mixing, Universal Grammar, and Second Language Acquisition.* In Ritchie, William and Bhatia, Tej eds. Handbook of Second Language Acquisition, 627-688, San Diego: Academic Press.

Bhatt, Rakesh 1997. Code-switching, Constraints, and Optimal Grammar. *Lingua* 102:223-251.

Goyal, P. et. al. 2003. Saarthak: A bilingual parser for Hindi, English and Code-switching Structures. *EACL Workshop: Computational Linguistics for South Asian Languages: Expanding Synergies with Europe*, April 12-17, Budapest.

Di Sciullo, A., Muysken, P., and Singh, R. 1986. Code-mixing and government. *Journal of Linguistics* 22:1-24.

Joshi, Aravind 1985. *Processing of Sentences with Intra-sentential Code Switching.* In Dowty, D. R., Karttunen, L., and Zwicky, A M. eds. Natural Language Parsing:Psychological, Computational, and Theoretical Perspectives, 190-205. Cambridge: Cambridge University Press.

Kachru, Braj 1978. Toward structuring code-mixing: An Indian perspective. *International Journal of the Sociology of Language* 16:28-46.

Muysken, P. C. 2000. *Bilingual Speech: Typology of Code-mixing.* Cambridge, Mass: Cambridge University Press.

Pfaff, Carol 1979. Constraints on Language Mixing. Language 55:291-319.

Poplack, Shana 2000. Code-switching. In Smelser, N. and Baltes, P. eds. International Encyclopedia of the Social and Behavioral Sciences, 2062-2065, Elsevier Science Ltd.

Sankoff, David and Poplack, Shana I981. A Formal Grammar for Code-switching. Papers in Linguistics 14:13-46.

Singh, Rajendra 1985. Grammatical Constraints on Code-mixing: Evidence from Hindi-English. Canadian Journal of Linguistics 30:33-45.

Sinha, R.M.K. 2004. An Engineering Perspective of Machine Translation: AnglaBharti-II and AnuBharti-II Architectures, Invited Paper, *Proceedings of International Symposium on Machine Translation, NLP and Translation Support System (iSTRANS- 2004)*, November 17-19, Tata Mc Graw Hill, New Delhi.

Sinha, R.M.K. 2001. Dealing with Unknown Lexicons in Machine Translation from English to Hindi, *Proc. IASTED International Conference on Artificial Intelligence and Soft Computing*, May 21-24, Cancun, Mexico, 333-336.

Sridhar, S. N. and Sridhar, K. 1980. The Syntax and Psycholinguistics of Bilingual Code Mixing. *Canadian Journal of Psychology* 35 (4):409-418.