Polibits

ISSN: 1870-9044

polibits@nlp.cic.ipn.mx

Instituto Politécnico Nacional

México

Mishra, Vimal; Mishra, R. B.

Study of Example Based English to Sanskrit Machine Translation

Polibits, vol. 37, 2008

Instituto Politécnico Nacional

Distrito Federal, México

Study of Example Based English to Sanskrit Machine Translation

Vimal Mishra and R. B. Mishra

***Abstract*—Example based machine translation (EBMT) has emerged as one of the most versatile, computationally simple and accurate approaches for machine translation in comparison to rule based machine translation (RBMT) and statistical based machine translation (SBMT). In this paper, a comparative view of EBMT and RBMT is presented on the basis of some specific features. This paper describes the various research efforts on Example based machine translation and shows the various approaches and problems of EBMT. Salient features of Sanskrit grammar and the comparative view of Sanskrit and English are presented. The basic objective of this paper is to show with illustrative examples the divergence between Sanskrit and English languages which can be considered as representing the divergences between the order free and SVO (Subject-Verb-Object) classes of languages. Another aspect is to illustrate the different types of adaptation mechanism.**

***Index Terms*—Example based machine translation, Devnagari, language divergence, matching.**

I. Introduction

THE Example Based Machine Translation (EBMT) is one of the most popular machine translation mechanisms which retrieve similar examples with their translation from the example data base and adapting the examples to translate a new source text. The origin of EBMT can be dated precisely to a paper by Nagao (1984). He has called the method "Translation by Analogy". The basic units of EBMT are sequences of words (phrases) and the basic techniques are the matching of input sentence (or phrases) with source example; phrase from the data base and the extraction of corresponding phrase from the data base and the extraction of corresponding translation (translation phrase) and the "recombination" of the phrases as acceptable translation sentences. It is defined on the basis of data used in translation process, and it is not enough to say that EBMT is "data driven" in contrast to "theory-driven" RBMT and that EBMT is "symbolic" in contrast to "non symbolic" SMT (John Hutchins, 2005). The emphasis is not on what matters but it is .how the data are used in translation operations (Turcato & Popowich, 1999).

Knowledge Driven Generalized EBMT system has been used which translates short single paragraph from English to Bengali (S. Bandyopadhyay, 2001). Headlines are translated using knowledge bases and example structures, while the sentences in the news body are translated by analysis and synthesis. In translation of news headlines, the various phrases in the source language and their corresponding translation in the target language are stored. The translations for the headlines are first searched in the table, organized under each headlines structure, containing specific source and target language pairs. If the headlines still can not be translated, syntax directed translation technique are applied. It matches with any phrase of a sentence structure and the bilingual dictionaries. Otherwise, word by word translation is attempted. The knowledge bases includes the suffix table for morphological analysis of English surface level words, parsing table for syntactic analysis of English, bilingual dictionaries for different classes of proper nouns, different dictionaries, different tables for synthesis in the target language.

One of the most remarkable basis that differentiate EBMT among RBMT and SMT is that the basic processes of EBMT are analogy-based , that is the search for phrases in the data base which are similar to input source language (SL) strings, their adaptation and recombination as target language (TL) phrases and sentences (Sumita *et al*.,1990). Neither RBMT nor SMT seek "similar" strings; both search for "exact" matches of input words and strings and produce sequence of words and strings as output. Thus, EBMT is analogy based MT while SMT is correlation based MT.

We have divided this paper into seven sections. Apart from introduction in section 1 the remaining sections are as follows. Section 2 discusses different approaches of EBMT like Foundation based approach, Run time approach, Template-Driven approach and Derivation based approach and then, we compare EBMT and RBMT (Rule Based Machine Translation) on basis of computational cost, improvement cost, system building cost, context-sensitive translation, robustness, measurement of reliability factor and example independency. Section 3 describes Sanskrit grammar, gives comparative view of English and Sanskrit language and discusses some previous work done on Sanskrit. Section 4 discusses different problems that occur in English to Sanskrit translation using EBMT. Section 5 covers different types of language divergences between English and Sanskrit. Section 6 discusses different adaptation technique used in EBMT

Manuscript received February 28, 2008. Manuscript accepted for publication June 13, 2008.

Vimal Mishra is with the Department of Computer Engineering, Institute of Technology, Banaras Hindu University (I.T.-BHU), Varanasi, India-221005 (Phone: +91-9415457592; e-mail: vimal.mishra.upte@gmail.com, vimal.mishra.cse07@itbhu.ac.in)

R. B. Mishra is with the Department of Computer Engineering, Institute of Technology, Banaras Hindu University (I.T.-BHU), Varanasi, India-221005 (e-mail: ravibm@bhu.ac.in).

system. Section 7 gives implementation steps to achieve the translation from English to Sanskrit. Section 8 draws the conclusions.

II. Approaches using EBMT and their Comparison

The existing EBMT system uses different approaches like Foundation based approach, Run time approach, Template-Driven approach and Derivation based approach. Then, we compare EBMT with RBMT as both are close to each other on some issues.

A. Approaches using EBMT

We can classify approaches that use EBMT into four categories as shown in figure 1 that are presented from the least rule based to the most rule based approach (John Hutchins, 2005).

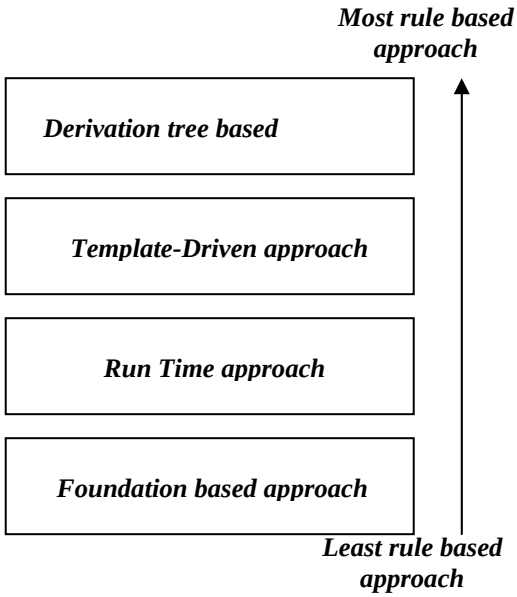

Fig. 1. Approaches based on EBMT

In Foundation based approach based on EBMT, the true EBMT systems are those where the information is not preprocessed, it is available and unanalyzed throughout the matching and execution processes.

In Run time approach using EBMT, (Planas & Furuse, 1999) EBMT uses a method of fuzzy matching involving superficial lemmatization and shallow parsing while E.Sumita *et al*. (1990) describe a full run time EBMT system that uses dynamic programming matching and thesauri for calculating semantic distances and illustrated by Japanese-English translation (at ATR in Japan).

In Template-Driven EBMT, methods of building templates from bilingual example corpora in advance of translation processes are used. Ilyas Cicekli & Altay Guvenir (1996) use templates in the form of words or lemmas with POS tags for a system with English as SL and Turkish as TL while Ralf Brown (2005) describes the induction of transfer rules in the form of templates of word strings, which are then either interpreted as rules of a transfer grammar or added as new examples to the original corpus.

Derivation trees approach of EBMT are devoted to the precompiled preparation of templates with more structure. Kaory Yamamoto & Yuji Matsumoto (1995) describe two studies extracting knowledge from an English-Japanese parallel corpus of business texts. The first study describes that word and phrase correspondences are derived using a statistical dependency parser and three variants are evaluated. The second study compares the statistical dependency model with methods using word segmentation (plain n-gram) and "chunk" boundaries; it is concluded that this method is most useful for preparing bilingual dictionaries in new domains(particularly for identifying compound nouns) while statistical dependency is most useful for disambiguation.

B. Comparison between EBMT and RBMT

We compare EBMT with RBMT on the different basis that shows the feature of EBMT which RBMT lacks as below in table I.

TABLE I
Comparison between EBMT and RBMT

Basis	EBMT	RBMT
Computational Cost	Low	High
Improvement Cost	Low	High
System Building Cost	Low	High
Context-Sensitive Translation	General architecture incorporating contextual information into example representation provides a way to translate context sensitively.	Needs another understanding device in order to translate context sensitively.
Robustness	Low; EBMT works on best match reasoning.	High; works on exact match reasoning.
Measurement of reliability factor	Yes; a reliability factor is assigned to the translation result according to the distance between input and retrieved similar example.	No; RBMT has no device to compute the reliability of the result.
Example Independency	Yes; knowledge is completely independent of the system, is usable in other system.	No; specific to a particular system.

III. Comparison of English and Sanskrit Grammar

English is well known language so we illustrate Sanskrit grammar and its salient features. The English sentence always has an order of Subject-Verb-Object, while Sanskrit sentence has a free word order. A free order language is a natural language which does not lead to any absurdity or ambiguity, thereby maintaining a grammatical and semantic meaning for every sentence obtained by the change in the ordering of the words in the original sentence. For example, the order of English sentence (ES) and its equivalent translation in Sanskrit sentence (SS) is given as below.

ES:	*Ram*	*reads*	*book.*
	(Subject)	*(Verb)*	*(Object)*
SS:	*Raamah*	*pustakam*	*pathati.*
	(Subject)	*(Object)*	*(Verb) ;* or
	Pustakam	*raamah*	*pathati.*
	(Object)	*(Subject)*	*(Verb) ;* or
	Pathati	*pustakam*	*raamah*
	(Verb)	*(Object)*	*(Subject)*

Thus Sanskrit sentence can be written using SVO, SOV and VOS order.

A. Alphabet

The alphabet, in which Sanskrit is written, is called Devnagari. The English language has twenty-six characters in its alphabet while Sanskrit has forty-two character or *varanas* in its alphabet. The English have five vowels (a, e, i, o and u) and twenty one consonants while Sanskrit have nine vowels or *swaras* (a, aa, i, ii, u, uu, re, ree and le) and thirty three consonants or *vyanjanas*. These express nearly every gradation of sound and every letter stands for a particular and invariable sound. The nine primary vowel consists of five simple vowel viz. a, i, u, re and le. The vowels are divided into two groups; short vowels: a, i, u, re and le and long vowels: aa, ii, uu, ree, lee, e, ai, o and au. Thus the vowels are usually given as thirteen. Each of these vowels may be again of two kinds: *anunasik* or nasalized and *ananunasik* or without a nasal sound. Vowels are also further discriminated into *udanta* or acute, *anudanta* or grave and *swarit* or circumflex. *Udanta* is that which proceeds from the upper part of the vocal organs. *Anudanta* is that which proceeds from their lower part while *Swarit* arises out of a mixture of these two. The consonants are divided into *sparsa* or mutes (those involving a complete closure or contact and not an approximate one of the organs of pronunciation), *antasuna* or intermediate (the semivowels) and *ilshman* or sibilants. The Consonants are represented by thirty three syllabic signs with five classes arranged as below.

(a) Mutes: (1) Kavarga: k, kh, g, gh, nn.
(2) Chavarga: ca, ch, j, jh, ni.
(3) Tavarga: t, th, d, dh, ne.
(4) Tavarga: t, th, d, dh, n.
(5) Pavarga: p, ph, b, bh, m.
(b) Semivowels: y, r, l, v.
(c) Sibilants: ss, sh, s.

The first two letters of the five classes and the sibilants are called surds or hard consonants. The rest are called sonants or soft consonants.

In Sanskrit, there are two nasal sounds: the one called *anuswara* and the other called *anunasika*. A sort of hard breathing is known as *visarga*. It is denoted by a sepcial sign: *a swara* or vowel is that which can be pronounced without the help of any other letter. A *vyanjana* or consonant is that which is pronounced with the help of a vowel.

B. Noun

According to Paninian grammar, declension or the inflections of the nouns, substantive and adjectives are derived using well defined principles and rules. The crude form of a noun (any declinable word) not yet inflected is technically called a *pratipadikai*.

C. *Gender*

Any noun has three genders: masculine, feminine, and neuter; three numbers: singular, dual, and plural. The singular number denotes one, the dual two and the plural three or more. The English language has two numbers: singular and plural, where singular denotes one and plural denotes two or more. There exist eight classifications in each number (grammar cases): nominative, vocative, accusative, instrumental, dative, ablative, genitive and locative. These express nearly all the relations between the words in a sentence, which in English are expressed using prepositions. Noun has various forms: *akAranta, AkAranta, ikAranta, IkAranta, nkAranta* and *makAranta*. Each of these *kaarakas*, have different inflections arising from which gender they correspond to. Thus, *akAranta* has different masculine and neuter declensions, *AkAranta* has masculine and feminine declensions, *ikAranta* has masculine, feminine and neuter declensions and *IkAranta* has masculine and feminine forms.

D. Pronoun

According to Paninian Grammar and investigations of M. R. Kale, Sanskrit has 35 pronouns. These pronouns have been classified into nine classes. Each of these pronouns has different classes as personal, demonstrative, relative, interrogative, reflexive, indefinitive, correlative, reciprocal and possessive. Each of these pronouns has different inflectional forms arising from different declensions of the masculine and the feminine form.

E. Adverb

Adverbs are either primitive or derived from noun, pronouns or numerals.

F. Particle

The particles are either used as expletives or intensive. In Sanskrit, particles do not possess any inflectional suffix, for example, *trata saa pathati*. Here, the word *trata* is a particle which has no suffix, yet the word *trata* implies the meaning of the seventh inflection.

G. Verb

There are two kinds of verbs in Sanskrit: primitive and derivative. There are six tenses (*Kaalaa*) and four moods

(*Arthaa*). The tenses are as present, aorist, imperfect, perfect, first future, and second future. The moods are as imperative, potential, benedictive and conditional. The ten tenses and moods are technically called the ten *Lakaras* in Sanskrit grammar.

H. Voice

There are three voices: the active voice, the passive voice and the impersonal construction. Each verb in Sanskrit, whether it is primitive or derivative, may be conjugated in the ten tenses and moods. Transitive verbs are conjugated in the active and passive voices and intransitive verbs in the active and the impersonal form. In each tense and mood, there are three numbers: singular, dual and plural with three persons in each.

I. Comparative View of English and Sanskrit

We describe comparative views of English and Sanskrit on different basis as below in table II.

TABLE II
COMPARATIVE VIEWS OF ENGLISH AND SANSKRIT

Basis	English	Sanskrit
Alphabet	26 character	42 character
Number of vowel	Five vowels	Nine vowels
Number of consonant	Twenty one consonant	Thirty three consonant
Number	Two: singular and plural	Three: singular, dual and plural
Sentence Order	SVO (Subject-Verb-Object)	Free word order
Tenses	Three: present, past and future	Six: present, aorist, imperfect, perfect. 1st future and 2nd future
Verb Mood	Five: indicative, imperative, interrogative, conditional and subjunctive	Four: imperative, potential, benedictive and conditional

Some previous works on Sanskrit are described below.

P. Ramanujan (1992) discusses the computer processing of the Sanskrit. Automatic morphological analysis should be performed. He also discusses syntactic, semantic and contextual analyses of Sanskrit sentence. In Sanskrit, words are composed of two parts: a fixed base part and a variable affix part. The variable part modifies the meaning of the word base, depending on a set of given relationships. The processes of declensions are properly defined. The Sanskrit is based on nominal stems, verbal stems and affixes. All available verbal stems are divided into ten specific classes (the Gana patha record groups of nominal stems, which undergo specific grammatical operations). There are 21 archetypal affixes for nominal declensions (denoted by '*sup'*) and 18 for verbs (denoted by '*tin'*). This is devised for the ending, gender etc. for noun (*subantas*) and for class (*gana*) a usage (*padi*). A nominal lexicon is then chosen, to cover all the allomorphic forms. The *Dhatupatha* is codified as verbal root lexicon. In semantic analysis, there are six functors, viz. Agent (5 types: independent doer causative agent, object agent (reflexive), expressed and unexpressed), object (7 types: accomplished, evolved, attained, desired, undesired, desired-undesired and agent-object), Instrument (2 types: internal and external), Recipient or Beneficiary (3 types: impelled, ascenting and non-refusing object. P. Ramanujan has developed a Sanskrit parser 'DESIKA', which is the analysis program based on Paninian grammar. DESIKA includes Vedic processing as well. In DESIKA, these are separate modules for the three functions of the system: generation, analysis and reference. Generation of nominal or verbal class of word is carried out by the user specifying the word and the applicable rules being activated. In analysis, the syntactic identification and assignment of functional roles for every word is carried out using the *Karaka-vibhakti* mappings. In the reference module, a complete 'trace' of the process of generation or analysis is planned to be provided, besides information or help. The DESIKA parser can be used by taking from the web http://www.tdil.mit.gov.in/download/desika.htm.

Rick Briggs (1985) uses semantic nets (knowledge representation scheme) to analyze sentences unambiguously. He compares the similarity between English and Sanskrit and the theoretical implications of this equivalence are given. In semantic nets, presentation of natural language object and subject is described in form of nodes, while relationship between them is described by edges. The meaning of the verb is said to be both *Vyapara* (action, activity, cause) and *Phulu* (fruit, result, effect). Syntactically, its meaning is invariably linked with the meaning of the verb "to do". All verbs have certain suffixes that express either the tense or mode or both, the person(s) engaged in the "action" and the number of persons or items so engaged.

IV. PROBLEMS IN ENGLISH TO SANSKRIT TRANSLATION USING EBMT

There are the following problems when we use the example based approaches to machine translation (Somer, 1999).

A. Parallel Corpora

EBMT is a corpus based MT, so this requires a parallel aligned corpus. The sources of machine readable parallel corpora are own parallel corpus of researchers, public domain parallel corpora. The EBMT system is generally to be best suited to a sublanguage approach and an existing corpus of translations can serve to define implicitly the sublanguage which the system can handle. When we use parallel aligned corpus from public domain, then the problem of sublanguage can arise. The parallel corpus, which is good enough, is quite difficult to get, especially for typologically different languages or for those languages that do not share the same writing system, such as English and Sanskrit. The alignment problem

of parallel corpus can be avoided by building the example database manually.

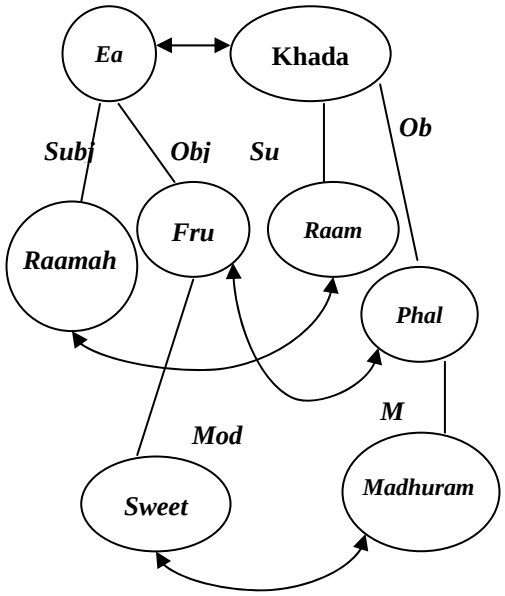

Fig. 2. Representation for English (E) and Sanskrit (S)

B. Granularity of Examples

The longer the matched passes, the probability of a complete match is the lower and the shorter the matched passes, the greater the probability of a complete match (Nirenburg *et al.*, 1993). The obvious and intuitive "grain size" for examples should be the sentence. Although the sentence as a unit for translation, offers the advantage such as sentence boundaries, are for the most part easy to determine.

C. Size of Example Database

There is a question: How many examples are needed in the example database to achieve the best translation result? According to Mima *et al.* (1998) the quality of translation is improved as more examples are added to the database. There is some limit after which further examples do not improve the quality of translation.

D. Suitability of Examples

According to Carl and Hansen (1999), a large corpus of naturally occurring text will contain overlapping examples of two types: (a) some examples will mutually reinforce each other, either by being identical, or by exemplifying the same translation phenomenon. (b) Other examples will be in conflict; the same or similar phrase in one language may have two different translations for no other reasons than inconsistency. According to Murata *et al*. (1999), the suitability of examples are taken by similarity metric, which is sensitive to frequency, so that a large number of similar examples will increase the score given to certain matches.

E. Structure of Examples Database

The structure of database with examples is concerned with storage of examples in the database, which is needed for searching the matches. In the simplest case, the examples may be stored as pairs of strings, with no additional information associated with them. As Somers and Jones (1992) point out, the examples might actually be stored with some kind of contextual manner. There is several structure of examples database of existing EBMT systems such as follows.

F. Annotated Tree Structures

In early EBMT systems, the examples are stored as fully annotated tree structures with explicit links. Figure 2 shows how the English example in E and Sanskrit Translation in S is represented. Similar ideas are found in Watanable (1992), Sato and Nagao (1990), Sadler (1991), Matsumoto *et al*. (1993), Sato (1995), Matsumoto and Kitamura (1995) and Meyers *et al*. (1998).

ES: Ram eats sweet fruits.
SS: Raamah madhuram phalam khaadati.
(Ram) (sweet) (fruits) (eats)

(Al-Adhaileh and Kong, 1999) examples are represented as dependency structures with links at the structural and lexical level expressed by indexes. Figure 3 shows the representation for the English-Sanskrit pair and figure 4 shows translation scheme for "*Shyam runs faster*".

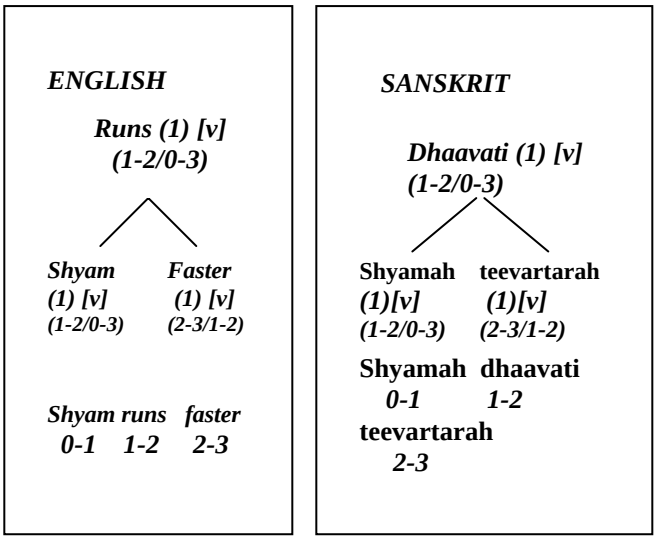

Fig. 3. Representation scheme for "*Shyam runs faster*"

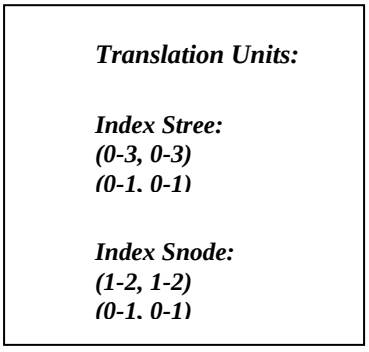

Fig. 4. Translation scheme for "*Shyam runs faster*"

ES: Shyam runs faster.
SS: Shyamah teevartarah dhaavati.
(Shyam) (faster) (runs)

The nodes in the trees are indexed to show the lexical head and span of the tree of which that item is head: so for the example the node labeled "*runs*" (1) [v] (1-2/0-3) indicates that the subtree headed by *runs*, which is the word spanning nodes 1 to 2 (i.e. the second word) is the head of the sub tree spanning nodes 0 to 3, i.e. *Shyam faster*. The labeled "Translation Units" gives the links between the two trees, divided into "Stree" links, identifying subtree correspondences (e.g. the English subtree 1-2 *runs* corresponds to the Sanskrit subtree *dhaavati* 1-2) and "Snode" links, identifying lexical correspondences (e.g. English word 1-2 *runs* corresponds to Sanskrit word 1-2 *dhaavati*).

G. Generalized Examples

In some systems, similar examples are combined and stored as a single "generalized" example. Brown (1999,) for instance, tokenizes the examples to show equivalence classes such as "person's name", "date", "city name" and also linguistic information such as gender and number. In Generalized Examples approach, phrases in the examples are replaced by these tokens, thereby making the examples more general.

H. Statistical Approach

In the statistical approach for structure of examples database, the examples are not stored at all, except in as much as they occur in the corpus on which the system is based (Somers, 1999).

I. Matching

The matching is a process that retrieves the similar examples from example data base. We describe some popular matching approaches below.

J. Character based Matching

The input sentence is matched with example sentence. The matching process involves a distance or similarity measure. When the examples are stored as strings, the measure may be a character-based pattern matching. In the earliest MT systems (ALPS "Repetitions processing" cf. Weaver, 1988), only exact matches of the alphanumeric strings were possible.

K. Word based Matching

Nagao (1984) proposed to use thesauri for indication of words similarity on the basis of meaning or usage. A thesaurus provides a listing of synonyms, allowing examples to match the input, on condition that they can be classified as synonyms based on a measurement of similarity. The examples in (1) and their translations in (2) (Nagao, 1984) show how this technique can be used successfully in choosing between conflicting examples.

(1) *(a) ES: A man eats vegetables.*
SS: Narah shaakam khaadati.
(A) (man) (vegetables) (eats)
(b) ES: Acids eats metal.
SS: Aambat dhaatum nashyati.
(Acids) (metal) (eats)

(2) *(a) ES: He eats potatoes.*
SS: Sah sukantham khaadati.
(He) (potatoes) (eats)
(b) ES: Sulphuric acid eats iron.
SS: Gandhak lauham nashyati.
(Sulphuric acid) (iron) (eats)

In 2 (a), the correct translation of *eats* (from Sanskrit translation SS) is chosen. This is correct in this instance as it refers to food and is chosen because of the relative similarity or distance between potatoes and vegetables.

L. Structure based Matching

In the earlier proposals for EBMT, it is assumed that the examples would be stored as structured objects, so the process involves a rather more complex tree-matching (e.g., Maruyama and Watanable 1992, Matsumoto *et al*. 1993, Watanable 1995, Al-Adhaileh and Tang 1999).

M. Annotated Word-based Matching

When we analyze both the input sentence and the examples to measure the similarity among them, then Annotated Word-Based Matching can be applied. Cranias *et al*. (1994, 1997) takes the function words for similarity measurement and makes use of POS tags. Veale and Way (1997) use sets of closed-class words to segment the examples which is said to be based on the "Marker Hypothesis" from psycholinguistics (Green, 1979).

N. Carroll's "Angle of similarity"

Carroll (1990) suggests the concept of an angle of similarity as a measure of distance between input sentence and the example sentence. This angle is calculated using a triangle whose three points represent the two sentences being compared and a 'null sentence'. The length of sides from this null point to the points representing the two sentences are the respective sizes of those sentences and the length of the third size is the difference between the two. The size of a sentence is calculated by costing the addition, deletion and replace operations necessary to derive one sentence from the other using costs from a set of 'rules' embodied in the system. We compare the given sentence with examples in the database looking for similar words and taking account of three basic operations. The relevance of particular mismatches is referred as "cost".

O. Partial Matching for Coverage

In most of the matching process, the aim is to find a single example or a set of individual examples that provide the best match for the input. In Nirenburg *et al* (1993), Somers *et al*. (1994) and Collins (1998), the matching process decomposes the cases and makes a collection of using terminology as "substring", "fragments" or "chunks" of the matched material. In these matching processes, the recombination process is needed for generating the target text (Jones, 1992: 165). If the dataset of examples is regarded as not a static set of discrete entities but a permutable and flexible interactive set of process modules, we can envisage a control architecture, where each process (example) attempts to close itself with respect to (parts of) the input.

P. Dynamic Programming Matching

Sumita (2003) applies an algorithm based on dynamic programming (DP) matching between word sequences for a speech to speech translation system. DP technique provides optimal solutions to specific problems by making decision at discrete time stages. At each stage, a small number of finite options are possible. Decisions are made, based on obtaining the optimal path from the input sentence to an example sentence. In Summit's approach, retrieval of examples is based on the calculation of a distance measure between the input and the example sentences. This distance measure is a normalized score of the sum of substitution, deletion and insertion operations. Once a similar example has been detected, the next step is to formulate a translation pattern from this example. These patterns are created dynamically and are not retained or stored for use in future translation.

Gelbukh and Sidorov (2006) show that dynamic programming gives least-cost hyper graph to formalize the paragraph alignment task in bilingual text such as English and Spanish. In formalization of the task, they select the optimal hyper graph out of hyper graphs with different number of arcs. Their algorithm prefers a smaller number of hyper arcs. It uses a (NE+1) (NS+1) chart, where NE and NS are the number of paragraphs in the text of the language English and Spanish, respectively. This algorithm has the complexity O (N^4), where N= NE = NS is the size of the text to be aligned.

Quirk and Menezes (2006) use dynamic programming for the dependency tree let translation that shows the convergence of statistical and example based machine translation. They have scored the head-relative positions of the tree as well as the root elements of the existing candidates. For the target-language model, we must multiply the probabilities of the neighbor words of each candidate. These additional probabilities depend only on a very small amount of information of the candidate. They have shown that dynamic programming does the search space savings, but it is not sufficient to produce a real-time translation system.

Q. Case Based Reasoning Matching

Case Based Reasoning (CBR) applies past cases to solve new problems. Each case contains a description of the problem and a possible solution. The Case–based ReVerb system (Collins, 1998) applies CBR technique to EBMT. In this approach, candidate examples are initially selected on condition that they share n words with the input. From this set, a parsed representation of each example is compared against a parsed representation of the input. This is an attempt to locate a match based on syntactic function. Syntactic function is combined with the additional parameters of sentence position and lexical equivalences. Where more than one match has been retrieved at this stage, matches are scored in terms of adaptability.

R. Boundary Friction Problem

The boundary friction is the problem of MT, when the same fragment of sentences needs inflections to indicate the grammatical case, such as determiner, adjective or noun. The boundary friction problem is difficult, in the case of language like Sanskrit, due to the fact that there is more than one grammatical inflection to indicate the syntactic function. So, for example, the translation associated with *the handsome boy* extracted, say, from (3), is equally reusable in the sentence (4,a), but it is not equally reusable in the sentence (4,b).

(3) ES: The handsome boy entered the room.
SS: Sundarah baalakah prakoshtam pravesham akarot.
(The) (handsome) (boy) (the) (room) (entered)

(4. a) ES: The handsome boy ate his breakfast.
SS: Sundarah baalakah svalapaahaaram agarhanaat.
(The) (handsome) (boy) (his) (breakfast) (ate)

(4. b) ES: I saw the handsome boy.
SS: Aaham sundaram baalakam apashyam.
(I) (the) (handsome) (boy) (saw)

S. Computational Problem

All the approaches of EBMT systems have to be implemented as software and significant computational factors influence many of them. One problem of such approaches, which stores the examples as complex annotated structures, is the huge computational cost in terms of creation, storage and matching or retrieval algorithms. This situation is problematic if such resources are difficult to obtain for one or both of the languages, as Guvenir and Cicekli (1998) report. Another problem of EBMT comes in picture when we extend the system's linguistic knowledge by increasing the size of example set (cf. Sato and Nagao, 1990:252). Adding more examples to the existing example database involves a significant overhead if these examples must be parsed and the resulting representations possibly checked by human. The next problem of EBMT is computational speed, especially for those of the EBMT systems that are used for real-time speech translation, which is solved by using "massively parallel processors".

V. Language Divergence between English and Sanskrit

Divergence is a common problem in translation between two natural languages. Language divergence (Dorr, 1993; Dave *et al*, 2001) occurs, when lexically and syntactically similar sentences of the source language are not translated into sentences that are similar in lexical and syntactic structure in the target language.

For example, consider the following English sentences and their Sanskrit translations:

(A) ES: She is in love.
SS: Saa madanesu asti.
(She) (love)(in) (is)

(B) ES: She is in train.
SS: Saa vaashpshakateshu asti.
(she) (train)(in) (is)

(C) ES: She is in fear.
SS: Saa vibheti.
(She) (is in fear)

Items (A) and (B) are examples of normal translation pattern. The prepositional phrases (PP) of the English sentences are similar to PP in Sanskrit though the prepositions occur after the corresponding noun in accordance with the Sanskrit syntax. Still example (C) has a structural variation.

The prepositional phrase “is in fear” is translated by the verb “*vibheti*”. This is an instance of a translation divergence.

We have considered that if the English sentence in (A) is given as the input to English to Sanskrit Example Base Machine Translation (EBMT) system, then two cases may arise:

1. The retrieved example is B, i.e., “She is in train”. In this case, the correct Sanskrit translation may be generated simply by using word replacement operation to replace “*vaashpshakateshu*” with “*madanesu*”.
2. If example (C) is retrieved for adaptation, the generated translation may be “*Saa* (she) *madaneshati* (love) (in) (is)”, which is syntactically incorrect Sanskrit sentence. So, the output of the system will depend entirely on the sentence (B), which will be retrieved to generate the translation of the input (A). We see that when we take example C to generate the translation of the input A, which gives us a syntactically incorrect Sanskrit sentence. This is due to the presence of divergence in the translation of example (C). Identification of divergence must be considered paramount for an EBMT system. So, an algorithm must be used in partitioning the example base into two parts: (i) divergence example base and (ii) normal example base.

This will help in efficient retrieval of past examples which improves the performance of an EBMT system.

VI. Divergence and its Identification: Some Relevant Previous Work

There are several approaches that deal translation divergence. We discuss some of them below.

A. Transfer Approach

In the transfer approach of translation divergence, there is transfer rule for transforming a source language (SL) sentence into target language (TL), by performing lexical and structural manipulations. These transfer rules are formed in several ways:

(i) With manual encoding (Han *et al*., 2000) and
(ii) With analysis of parsed aligned bilingual corpora (Watanable *et al*, 2000).

B. Interlingua Approach

In the interlingua approach, the identification and resolution of divergence are based on two mappings GLR (Generalized Linking Routine), CSR (Canonical Syntactic Realization) and a set of LCS (Lexical Conceptual Structure) parameters. The translation divergence occurs, when there is an exception either to GLR or to CSR (or to both) in one of the languages. This situation permits one to formally define a classification of all possible lexical-semantic divergences that could arise during translation. This approach has been used in the UNITRON system (Dorr, 1993) that performs translation from English to Spanish and English to German.

C. Generation Heavy Machine Translation (GHMT) Approach

The MATADOR System (Habash, 2003) uses this approach for translation between Spanish and English. In this approach, a symbolic overgeneration is created for a target glossed syntactic dependency representation of SL sentences, which uses rich target language resources, such as word-lexical semantics, categorical variations and sub-categorization frames for generating multiple structural variations. This is constrained by a statistical TL model that accounts for possible translation divergences. Then, a statistical extractor is used for extracting a preferred sentence from the word lattice of possibilities. This approach bypasses explicit identification of divergence, and generates translations, which may include divergence sentences otherwise.

D. Universal Networking Language based Approach

In Universal Networking Language (UNL), sentences are represented using hypergraphs with concepts as nodes and relations as directed arcs. A dictionary of UW (Universal Word) is maintained. A divergence is said to occur if the UNL expression generated from the both source and target language analyzer differ in structure. Dave *et al* (2002) proposed UNL approach for English to Hindi machine translation.

Each of the above approaches has problems, when we apply them in English to Sanskrit machine translation. For example, GHMT (Generation Heavy Machine Translation) approach requires rich resources for the target language (here, Sanskrit), which is not available for Sanskrit nowadays. The Interlingua approach requires deep semantic analysis of the sentences and creation of exhaustive set of rules to capture all the lexical and syntactic variation may be problem in English to Sanskrit translation. While in case of UNL based approach, each UW of the dictionary contains deep syntactic, semantic and morphological knowledge about the word. Creation of such UW dictionary for a restricted domain is difficult and rarely happens.

With respect to Sanskrit, the major problem in applying the above approach is that linguistic resources are very scarce for Sanskrit.

We proposes an approach that uses only the functional tags (FT) and syntactic phrasal annotated chunk (SPAC) structures of the source language (SL) and target language (TL) sentences for identification of divergences. In a translation example, a translation divergence occurs when some particular FT upon translation is realized with the help of some other FT in the target language. The occurrence of divergence is identified by comparing different constraints of words in the source and target language sentence.

VII. Divergences and its Identification in English to Sanskrit Translation

Divergence is a language dependent phenomenon, it is not expected that the same set of divergences will occur across all languages. Dorr (1993) classifies divergence in seven broad types, which is lexical-semantic divergences for translating among the European languages, as below.

(i) Structural divergence
(ii) Conflational divergence
(iii) Categorial divergence
(iv) Promotional divergence
(v) Demotional divergence
(vi) Thematic divergence

(vii) Lexical divergence

A. Structural Divergence

A structural divergence is said to have occurred if the object of the English sentence is realized as a noun phrase (NP) but upon translation in Sanskrit it is realized as a prepositional phrase (PP). The following examples illustrate this.

(a) ES: Ram will attend this meeting.
SS : Ramah asyaam sabhaayaam anuvartishyate.
(Ram) (this) (meeting in) (will attend)

(b) ES: Ram married Sita.
SS: Ramah Sitayaa sahpaanigrahanam akarot.
(Ram) (Sita)(with) (married)

(c) ES: Ram will challenge Mohan.
SS: Ramah Mohanam aahanyashyate.
(Ram) (Mohan) (will challenge).

Analysis of above examples gives us the following points with respect to structural divergence, which we use to design the algorithm for identification of structural divergence.

(i) If the main verb of an English sentence is a declension of "be" verb, then the structural divergence cannot occur.
(ii) Structural divergence deals with the objects of both the English sentence and its Sanskrit translation. So, if any one of the two sentences has no objects then structural divergence cannot occur.
(iii) If both sentences have objects, and then SPAC structures are same then also structural divergence does not occur.
(iv) In this situation, structural divergence may occur only if the SPAC of the object of the English sentence is an NP, and the SPAC of the object of the Sanskrit sentence is a PP.

B. Categorial Divergence

If English sentence has subjective complement (SC) or predictive adjustment (PA), then categorical divergence occurs. In the categorical divergence, the SC or PA of the English sentence, upon translation, is realized as the main verb of the Sanskrit sentence. The SC may be noun phrase (NP) or adjective phrase (AdjP) and PA may be prepositional phrase (PP) or adverb in the English sentence. The categorial divergence is concerned with adjectival SCs which upon translation map into noun, verb or PP. In English to Sanskrit translation, depending upon the nature of the SC or PA, the following subtypes of categorial divergence have been identified, which are given below.

(i) Categorial Subtype 1

When the SC of the English sentence is used as an adjective, but upon translation, it is realized as the main verb of the Sanskrit sentence, then this divergence occurs. For example, consider the following sentences given below.

ES: Ram is afraid of lion.
SS: Ramah singhaat vibheti.
(Ram) (of) (lion) (afraid)

The adjective of the English sentence "afraid" is realized in Sanskrit by the verb "*vibh*" meaning "afraid" and "*vibheti*" is it's conjugate form for present indefinite tense, when the subject is first person, singular and masculine in Sanskrit.

(ii) Categorial subtype 2

When the SC is an NP in the English sentence, then after translation the noun part corresponds to the verb of the Sanskrit sentence. This part is realized as an adverb upon translation.

Consider the following sentences given below.

ES: Ram is a regular user of the library.
SS: Ramah pustakaalayasya aharvisham prayogam karoti.
(Ram) (library)(of) (regular) (user)

The word "user", which is a noun, has been used as an SC in the English sentence above. This provides the main verb "*prayogam karoti*" (meaning "to use") of the Sanskrit sentence. The adjective "regular" of the noun "user" is realized as the adverb "*aharvisham*".

(iii) Categorial subtype 3

The adverbial PA of an English sentence is realized as the main verb of the Sanskrit sentence, for example,

ES: The fan is on.
SS: Vyajanam chalati.
(fan) (move) (ing) (is)

The main verb of the Sanskrit is "*chal*" i.e. "to move". Its sense comes from the adverbial PA "on" of the English sentence. The present continuous form of this verb is "*chalati*", when the subject is third person, singular and masculine in Sanskrit.

(iv) Categorial subtype 4

The PA that is realized in English as PP, but PA is realized in Sanskrit as the main verb. For example, consider the following sentences given below.

ES: The train is in motion.
SS: Railyaanam chalatii.
(train) (move) (ing) (is)

The PA "in motion" is a preposition phrase which sense is realized by the verb "*chal*". In Sanskrit translation, the present continuous form of this verb is "*chalati*", because the subject of the sentence is feminine and singular. After the analysis of these translation examples, we get the following cases related to above mentioned ones..

(i) Categorial divergence occurs if the main verb of the English sentence is a declension of "be" but the main verb of the Sanskrit translation is not the "be" verb.
(ii) Categorial divergence occurs if the Sanskrit translation does not have any subjective complement or PA.
(iii) If SPAC structure of the SC of English sentence is an AdjP or NP then categorial divergence will be of subtype 1 or 2, respectively.
(iv) If SPAC structure of PA of English sentence is AdvP or PP then categorial divergence will be of subtype 3 or 4, respectively.

C. Nominal Divergence

Nominal divergence is concerned with the subject of the English sentence. After translation, the subject of the English sentence becomes the object or verb complement. This nominal divergence is similar to thematic divergence of Dorr (1993).

The subject of the English sentence is realized in Sanskrit with the help of a prepositional phrase. We define two subtypes of nominal divergence as below.

(i) Nominal subtype 1

The subject of the English sentence becomes object upon Translation. For example, consider the following sentences.

ES: Ram is feeling hungry.
SS: Raamen ksudhitaa anubhuuyate.
(To Ram) (hunger) (feeling) (is)

The adjective "hungry" is an SC. Its sense is realized in Sanskrit by the word "*ksudhita*" that acts as the subject of the Sanskrit sentence. The subject "Ram" of the English sentence becomes the object "*Raamen*" (to Ram) of the Sanskrit translation.

(ii) Nominal subtype 2

The subject of the English sentence provides a verb complement (VC) in the Sanskrit translation. For example, consider the following sentences below.

ES: This gutter smells foul.
SS: Asmaat jalanirgamaat malinam jighrati.
(This) (gutter)(from) (foul) (smells)

The subject of the English sentence "This gutter" is realized as the modifier "*Asmaat jalanirgamaat*" of the verb "*anubhavati*".

The analysis of above examples gives the following points.

(i) If the English sentence does not have an SC or declension of the "be" verb, then divergence is to be nominal.

(ii) If the SC of English sentence is null and the object is not null in Sanskrit then it is the instance of nominal divergence of subtype 1. If verb complement (VC) is present in Sanskrit then it nominal divergence of subtype 2.

D. Pronominal Divergence

Pronominal divergence occurs if the pronoun "it" is used as the subject in English sentences. The Sanskrit equivalent of "it" is "*edam*". So, the Sanskrit translation of such a sentence should have "*edam*" as the subject of the sentence. For example, consider the following sentences.

ES: It is crying.
SS: Edam krandati.
(It) (is crying)
ES: It is small.
SS: Edam laghu asti
(It) (is small)

E. Demotional Divergence

When the main verb of the English sentence upon translation is demoted to the subjective complement or predicative adjunct of the Sanskrit sentence and the main verb of Sanskrit translation are realized as "be" verb, then demotional divergence occurs. For example, consider the following sentences.

ES: This house belongs to a doctor.
SS: Edam griham ekasya chikitsakasya asti.
(This) (house) (one) (doctor) (of) (is)
ES: This dish feeds four people.
SS: Edam bhojanam chaturthajanebhyah asti.
(this) (dish) (four) (people) (for) (is)

F. Conflational Divergence

The conflational divergence pertains to the main verb of the source language sentence. According to Dorr (1993), the conflational divergence occurs, when some new words are required to be incorporated in the target language sentence in order to convey the proper sense of a verb of the input.

G. Possessional Divergence

The possessional divergence occurs when the verb "have" in the English sentence is used as the main verb. For example, consider the following sentences given below.

ES: Mohan has many enemies.
SS: Mohanasya anekaah shatrvah santi.
(Mohan) (many) (enemies) (has)

VIII. ADAPTATION

After matching and retrieval of a set of examples, with associated translations, the next step in the EBMT systems is to extract from the translations, the appropriate fragments ("alignment" or "adaptation") and combine these fragments so as to produce a grammatical target output, which is called as recombination. These processes are carried out as twofold that is identifying which fragment of the associated translation corresponds to the matched fragments of the source text and recombining these fragments in an appropriate manner. We can illustrate the problem by considering English to Sanskrit translation below.

1. *(a) ES: He buys a notebook.*
   SS: Sah ekaah panjikam krinaati.
   (b) ES: He read a book on Hindi.
   SS: Sah ekam Hindyaam pustakam pathati.
   (c) ES: He buys a book on Hindi.
   SS: Sah ekam Hindyaam pustakam krinaati.

To understand how the relevant elements of (1: a, b) are combined to give (1, c), we must assume a mechanism to extract from them the common elements (underlined here). Then, we have to make the further assumption that they can be simply pasted together as in (1, c) and that this recombination will be appropriate and grammatical.

The need for an efficient systematic adaptation scheme is required for modifying a retrieved example and thus, generating the required translation. Some of major adaptation approaches of an EBMT system are described below.

(1) Veale *et al*. (1997) proposed adaptation in Gaijian via two categories: high-level grafting and key hole surgery. The phrases are handled with high level grafting. In the high level grafting, an entire phrasal segment of the target sentence is replaced with another phrasal segment from a different example. The key hole surgery deals with individual words in an existing target segment of an example. Under the key hole surgery operation, words are replaced to fit the current translation task. For example, suppose the input sentence is "The girl is playing in the lawn", and in the example base, we have the following examples.

(a) The child is playing.
(b) Sita knows that girl.
(c) It is a big lawn.
(d) Shyam studies in the school.

The sentences (a) and (d) will be used for high level grafting. Then key hole surgery will be applied for putting in the translations of the words "lawn" and "girl". These translations will be extracted from (b) and (c).

(2) In Shiri *et al*. (1997), adaptation procedure is based on three steps: finding the difference, replacing the difference and smoothing the output. The differing segments of the input sentence and the source template are identified. The translations of these different segments in the input sentence are produced by rule-based methods and these translated segments are fitted into a translation template. The resulting sentence is then smoothed over by checking for person, and number agreement and inflection mismatches. For example, assume the input sentence and selected templates as below.

SI : A very efficient lady doctor is busy.

ST : A lady doctor is busy.

TT: Ekaa mahilaa chikitsaka kaaryavyagrah asti.

The parsing process shows that "*A very efficient lady doctor*" is a noun phrase and so matches it with "*A lady doctor*" ("*Ekaa mahilaa chikitsaka*"). "*A very efficient lady doctor*" is translated as "*Ekaa bahut yogyah mahilaa chikitsaka*", by rule based noun phrase translation system. This is inserted into TT giving the following TT: *Ekaa bahuyogyah mahilaa chikitsaka kaaryavyagrah asti*.

(3) Collins (1998) proposed the adaptation scheme as ReVerb system. In this, two different cases are considered: Full case adaptation and Partial case adaptation. Full case adaptation is used when a problem is fully covered by the retrieved example and desired translation is created by substitution alone. In Full case adaptation, five scenarios are possible that are SAME, ADAPT, IGNORE, ADAPT_ZERO and IGNORE_ZERO. Partial case adaptation is used when a single unifying example does not exist. In this case, three more operations are required on the top of the above five. These three operations are ADD, DELETE and DELETE_ZERO.

(4) Somers (2001) proposed adaptation scheme that uses case based reasoning (CBR). The simplest of the CBR adaptation method is null adaptation, where no changes are recommended. In a more general situation, various substitution methods (e.g. Reinstantiation, Parameter Adjustment), transformation methods (e.g. Commonsense transformation and model-guided repair) may be applied. For example, suppose the input sentence (I) and the retrieve examples (R).

I: That old woman has come.

R: That old man has come. (vrddhah aagacchat.)

To generate the desired translation of the word "*man*" ("*vrddhah*") is first replaced with the translation of "*woman*" ("*vrddhaah*") in R. This operation is called reinstitution. At this stage, an intermediate translation "*vrddhah aagacchat*" is obtained.

(5) Jain (1995) proposed HEBMT system, in which examples are stored in an abstracted form for determining the structural similarity between the input sentence and the example sentences. The target language sentence is generated using the target pattern of the sentence that has lesser distance with the input sentence. The system substitutes the corresponding translations of syntactic units identified by a finite state machine in the target pattern. Variation in tense of verb and variations due to number, gender etc. are taken care at this stage for generating the appropriate translation. The HEBMT system translates from Hindi to Sanskrit.

Thus in our view, the adaptation procedures employed in different EBMT systems primarily consists of four operations that are given as below.

(i) Copy: Where the same chunk of the retrieved translation example is used in the generated translation.

(ii) Add: Where a new chunk is added in the retrieved translation example.

(iii) Delete: When some chunk of the retrieved example is deleted and

(iv) Replace: Where some chunk of the retrieved example is replaced with a new one to meet the requirement of the current input.

IX. Implementation

Each translation example record in our example base contains morpho-functional tag information for each of the constituent word of the source language (English) sentence, its Sanskrit translation and the root word correspondence. These tags are obtained by the ENGCG parser (http://www.lingsoft.fi/cgi-bin/engcg) for English sentences. The Sanskrit parser is obtained from the Sanskrit heritage site (http://sanskrit.inria.fr/) which is developed by Gerard Huet. The English to Sanskrit on line dictionary is taken from the site (www.dicts.info/dictionary.php?l1=English&l2=Sanskrit).

X. Conclusions

The EBMT is "data driven" in contrast to "theory driven" RBMT, which retrieves similar examples (pairs of source sentences and their translations), adapting the examples to translate a new source sentence. The Example-Based Machine Translation is used in situations, where on-line resources (such as parser, morphological analyzer, rich bilingual dictionary, rich parallel corpora, etc) are scarce. The Sanskrit is free word order language. Thus, we maintain a grammatical and semantic meaning for every sentence obtained by the change in the ordering of the words in the original sentence. The language divergence significantly occurs between English and Sanskrit translation. Suitable illustrations through examples for some popular adaptation approaches have been given. The adaptation processes select the best match of example sentences and suggests the adaptation procedures employed in different EBMT systems primarily consists of four operations: copy, add, delete and replace. The basic objective of the paper is to illustrate with examples the divergence and adaptation mechanism in English to Sanskrit.

References

[1] M. H. Al-Adhaileh and E. K. Kong, "Example-Based Machine Translation Based on the Synchronous SSTC Annotation Schema", Machine Translation Summit VII, Singapore, pp. 244–249, 1999.

[2] S. Bandyopadhyay, "An Example Based MT System in News Items Domain from English to Indian Languages", Machine Translation Review 12, pp 7-10, 2001.

[3] Rick. Briggs, "Knowledge Representation in Sanskrit and Artificial Intelligence", The AI Magazine, pp 33–39, 1985.

[4] R. D. Brown, "Adding Linguistic Knowledge to a Lexical Example-based Translation System", in TMI, pp. 22–32, 1999.

[5] R. Brown, "Context-sensitive retrieval for example-based translation", MT Summit X, Phuket, Thailand, September 16, Proceedings of Second Workshop on Example-Based Machine Translation, pp. 9-15, 2005.

[6] M. Carl, “Inducing Translation Templates for Example-Based Machine Translation”, Machine Translation Summit VII, Singapore, pp. 250–258, 1999.
[7] J. J. Carroll, “Repetitions Processing Using a Metric Space and the Angle of Similarity”, Report No. 90/3, Centre for Computational Linguistics, UMIST, Manchester, England, 1990.
[8] Cicekli and H. A. Güvenir, “Learning Translation Rules From A Bilingual Corpus”, NeMLaP-2: Proceedings of the Second International Conference on New Methods in Language Processing, Ankara, Turkey, pp. 90–97, 1996.
[9] B. Collins, “Example-Based Machine Translation: An Adaptation-Guided Retrieval Approach”, PhD thesis, Trinity College, Dublin, 1998.
[10] L. Cranias, H. Papageorgiou and S. Piperidis, “A Matching Technique in Example-Based Machine Translation”, in Coling, pp. 100–104, 1994.
[11] L.Cranias, H. Papageorgiou and S. Piperidis, “Example Retrieval from a Translation Memory”, Natural Language Engineering 3, 255–277, 1997.
[12] Dave, Sachi, Parikh, Jignashu, Pushpak Bhattacharya, “Interlingua-based English–Hindi Machine Translation and Language Divergence”, Machine Translation, Vol. 16, pp. 251–304, Kluwer Academic Publishers, 2001.
[13] B. J. Dorr, “Machine Translation: A View from the Lexicon”, MIT Press, Cambridge, MA, 1993.
[14] Bonnie J. Dorr, “Machine Translation Divergences: A Formal Description and Proposed Solution”, Association for Computational Linguistics, pp 597-633, 1994.
[15] Nano Gough, “Example-Based Machine Translation using the Marker Hypothesis”, PhD thesis, School of Computing, Dublin, 2005.
[16] T. R. G. Green, “The Necessity of Syntax Markers: Two Experiments with Artificial Languages”, Journal of Verbal Learning and Verbal Behavior 18, 481–496, 1979.
[17] H. A. Guvenir and I. Cicekli, “Learning translation templates from examples”, Information System 23, 353-363, 1998.
[18] Deepa Gupta, “Contributions to English to Hindi Machine Translation Using Example-Based Approach”, PhD thesis, IIT Delhi, 2005.
[19] N. Habash, “Generation-Heavy Hybrid Machine Translation”, PhD thesis, University of Maryland, College Park, 2003.
[20] C. Han, L. Benoit, P. Martha, R. Owen, R. Kittredge, T. Korelsky, N. Kim and M.Kim, “Handling structural divergences and recovering dropped arguments in a Korean to English machine translation system”, Proceedings of the Fourth Conference of the Association for Machine Translation in the Americas, AMTA-2000, Cuernavaca, Mexico, 2000.
[21] J. Hutchins, “Towards a Definition of Example-Based Machine Translation”, In Machine Translation Summit X, Second Workshop on Example-Based Machine Translation, pages 63–70, Phuket, Thailand 2005.
[22] J. Hutchins, “Example-Based Machine Translation: A Review and Commentary”, Machine Translation, Vol. 19, pp 197-211, 2005.
[23] R. Jain, “HEBMT: A Hybrid Example-Based Approach for Machine Translation (Design and Implementation for Hindi to English)”, PhD thesis, I.I.T. Kanpur, 1995.
[24] D. Jones, “Non-hybrid example-based machine translation architectures”, In TMI, pp. 163-171, 1992.
[25] M. R. Kale, “A Higher Sanskrit Grammar”, 4th Ed, Motilal Banarasidas Publishers Pvt. Ltd., 2005.
[26] Macdonnel, “A Sanskrit Grammar for Students”, 3rd Ed, Motilal Banarasidas Publishers Pvt. Ltd, 2003.
[27] Maruyama and H. Watanabe, “Tree Cover Search Algorithm for Example-Based Translation”, in TMI, pp. 173–184, 1992.
[28] Y. Matsumoto, H. Ishimoto and T. Utsuro, “Structural Matching of Parallel Texts”, 31st Annual Meeting of the Association for Computational Linguistics, Columbus, Ohio, pp. 23–30, 1993.
[29] Y. Matsumoto and M. Kitamura, “Acquisition of Translation Rules from Parallel Corpora”, in Mitkov & Nicolov, pp. 405–416, 1995.
[30] Meyers, R. Yangarber, R. Grishman, C. Macleod and A. Moreno-Sandeval, “Deriving Transfer Rules from Dominance-Preserving Alignments”, in Coling-ACL, pp. 843–847, 1998.
[31] H. Mima, H. Iida and O. Furuse, “Simultaneous Interpretation Utilizing Example-based Incremental Transfer”, in Coling-ACL, pp. 855–861, 1998.
[32] M. Murata, Q. Ma, K. Uchimoto and H. Isahara, “An Example-Based Approach to Japanese to English Translation of Tense, Aspect, and Modality”, in TMI, pp. 66–76, 1999.
[33] M. Nagao, “A Framework of a Mechanical Translation between Japanese and English by Analogy Principle”, in A. Elithorn and R. Banerji (eds), Artificial and Human Intelligence, Amsterdam: North-Holland, pp. 173–180, 1984.
[34] Chakradhar Nautiyal, “Vrihad Anuvaad Chandrika”, 4th Ed., Motilal Banarasidas Publishers Pvt. Ltd, 1997.
[35] S. Nirenburg, C. Domashnev and D. J. Grannes, “Two Approaches to Matching in Example-Based Machine Translation”, in TMI, pp. 47–57, 1993.
[36] E. Planas and O. Furuse, “Formalizing Translation Memories”, Machine Tran Summit VII, Singapore, pp. 331–339, 1999.
[37] P. Ramanujan, “Computer Processing Of Sanskrit”, Computer Processing Of Asian Languages CALP-2, IIT Kanpur, 1992.
[38] V. Sadler, “The Textual Knowledge Bank: Design, Construction, Applications”, International Workshop on Fundamental Research for the Future Generation of Natural Language Processing (FGNLP), Kyoto, Japan, pp. 17–32., 1991.
[39] S. Sato and M. Nagao, “Toward Memory-Based Translation”, in Coling, Vol. 3, pp. 247–252, 1990.
[40] S. Sato, “MBT2: A Method for Combining Fragments of Examples in Example-Based Machine Translation”, Artificial Intelligence 75, 31–49, 1995.
[41] S. Shiri, F. Bond and Y. Takhashi, “A Hybrid Rule and Example-Based Method for Machine Translation”, Proceedings of the 4th Natural Language Processing Pacific Rim Symposium: NLPRS-97, Phuket, Thailand, pp. 49-54, 1997.
[42] H. Somers, “Review article: example-based machine translation”, Machine Translation 14 (2), 113-157, 1999.
[43] H. Somers, and D. Jones, “Machine Translation Seen as Interactive Multilingual Text Generation”, Translating and the Computer 13: The Theory and Practice of Machine Translation – A Marriage of Convenience?, London, Aslib, pp. 153–165, 1992.
[44] H. Somers, I. McLean and D. Jones, “Experiments in Multilingual Example-Based Generation”, CSNLP 1994: 3rd Conference on the Cognitive Science of Natural Language Processing, Dublin, Ireland, 1994.
[45] H. Somers, “EBMT seen as case-based reasoning”, MT Summit VIII Workshop on Example-Based Machine Translation, Santiago de Compostela, Spain, pp. 56-65, 2001.
[46] E. Sumita, H. Iida and H. Kohyama, “Translating with Examples: A New Approach to Machine Translation”, in TMI, pp. 203–212, 1990.
[47] E. Sumita, “EBMT Using DP-matching Between Word Sequences”, In Recent Advances in Example-Based Machine Translation, M. Carl and A. Way, Ed., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003, pp. 189–209.
[48] T. Veale and A. Way, “Gaijin: A Bootstrapping Approach to Example-Based Machine Translation”, International Conference, Recent Advances in Natural Language Processing, Tzigov Chark, Bulgaria, pp. 239–244, 1997.
[49] H. Watanabe, “A Similarity-Driven Transfer System”, in Coling, pp. 770–776, 1992.
[50] H. Watanabe, “A Model of a Bi-Directional Transfer Mechanism Using Rule Combinations”, Machine Translation 10, 269–291, 1995.
[51] H. Watanabe, S. Kurohashi and E. Aramaki, “Finding structural correspondences from bilingual parsed corpus for Corpus-Based Translation”, Proceedings of COLING, Saarbrucken, Germany, 2000.
[52] Weaver, “Two Aspects of Interactive Machine Translation”, in Technology as Translation Strategy, M. Vasconcellos Ed., Binghamton, NY: State University of New York at Binghamton (SUNY), 1988, pp. 116–123.
[53] Alexander Gelbukh and Grigori Sidorov, “Alignment of Paragraphs in Bilingual Texts using Bilingual Dictionaries and Dynamic Programming”, in Lecture Notes in Computer Science, N 4225, ISSN 0302-9743, Springer-Verlag, 2006, pp 824-833.
[54] C. Quirk and A. Menezes, “Dependency tree let translation: the convergence of statistical and example-based machine-translation?”, Journal of Machine Translation, Vol. 20, pp. 43–65, Kluwer Academic Publishers, 2006.