Ontology Based Machine Translation for Bengali as Low-resource Language.

KHAN MD. ANWARUS SALAM

A thesis submitted in partial fulfillment of
the requirements for the degree of
DOCTOR OF PHILOSOPHY

DEPARTMENT OF INFORMATION AND COMMUNICATION ENGINEERING
THE UNIVERSITY OF ELECTRO-COMMUNICATIONS
MARCH 2014

JAPANESE ABSTRACT

本研究では、WordNet と UNL オントロジーを用いた、オントロジーに基づく機械翻訳を提案する。ベンガル語のような低資源言語 (low-resource language)に対しては、具体例に基づく機械翻訳 (EBMT)は、あまり有効ではない。パラレル・コーパスの欠如のために、多数の未知語を扱わなければならなくなるためである。

我々は、低資源言語間の EBMT システムを実装した。実装した EBMT アーキテクチャでは、chunk-string templates (CSTs)と、未知語翻訳メカニズムを用いている。CST は、起点言語のチャンク、目的言語の文字列と、単語アラメント情報から成る。CST は、英語チャンカーを用いて、アラインメント済みのパラレル・コーパスと WordNet から、自動的に生成される。

最初に、起点言語のチャンクが OpenNLP チャンカーを用いて自動生成される。そして、初期 CST が、各起点言語のチャンクに対して生成され、すべての目的文に対する CST アラインメントがパラレル・コーパスを用いて生成される。その後、システムは、単語アラインメント情報を用いて、CST の組合せを生成する。

最後に、WordNet を用いて、広い適用範囲を得るために CST を一般化する。未知語翻訳に対しては、WordNet hypernym tree と、英語・ベンガル語辞書を用いる。提案システムは、最初に、未知語に対して、WordNet から意味的に関連した英単語を発見しようと試みる。これらの関連語から、英語・ベンガル語辞書にベンガル語の翻訳が存在する、意味的に最も近い語を選ぶ。もし、ベンガル語の翻訳が存在しなければ、システムは IPA-based 翻訳を行う。固有名詞に対しては、システムは、Akkhor 翻訳メカニズムを用いる。

CST は 57 ポイントの広い適用範囲を持つように改善され、その際の人間による訳文の評価も 48.81 ポイントを得た。現在、システムのよって、64.29%のテストケースの翻訳が行える。未知語メカニズムは、人間に評価において 3.56 ポイント、翻訳の質を改善した。CST と未知語の組合せよる解法は、テストケースにおいて、67.85%の許容可能な翻訳を生成した。

また、本研究では、UNL オントロジーが提供する semantic background を用いて、各概念に対する説明を自動生成する方法も提案した。このシステムに対する入力は、１つのユニバーサル・ワード(UN)であり、システムの出力はその UN の英語や日本語による説明文である。

与えられた UN に対して、システムは、最初に、SemanticWordMap を発見するが、それは、１つの特定の UN に対する、UNL オントロジーからのすべての直接的、間接的参照関係を含む。したがって、このステップの入力は、１つの UN であり、出力は WordMap グラフである。次のステップで、変換規則を用いて、WordMap グラフを UNL に変換する。この変換規則は、ユーザの要求に応じて、“From UWs only”や “From UNL Ontology” と指定できる。したがって、このステップの入力は WordMap グラフであり、出力は UNL 表現である。最終ステップでは、UNL DeConverter を用いて UNL 表現を変換し、自然言語を用いて記述する。これらの表現は、未知語に対する翻訳の質の向上に有効であることがわかった。

ABSTRACT

In this research we propose ontology based Machine Translation with the help of WordNet and UNL Ontology. Example-Based Machine Translation (EBMT) for low resource language, like Bengali, has low-coverage issues. Due to the lack of parallel corpus, it has high probability of handling unknown words. We have implemented an EBMT system for low-resource language pair. The EBMT architecture use chunk-string templates (CSTs) and unknown word translation mechanism. CSTs consist of a chunk in source-language, a string in target-language, and word alignment information. CSTs are prepared automatically from aligned parallel corpus and WordNet by using English chunker. For unknown word translation, we used WordNet hypernym tree and English-Bengali dictionary. Proposed system first tries to find semantically related English words from WordNet for the unknown word. From these related words, we choose the semantically closest related word whose Bangla translation exists in English-Bangla dictionary. If no Bangla translation exists, the system uses IPA-based-transliteration. For proper nouns, the system uses Akkhor transliteration mechanism. CSTs improved the wide-coverage by 57 points and quality by 48.81 points in human evaluation. Currently 64.29% of the test-set translations by the system were acceptable. The combined solutions of CSTs and unknown words generated 67.85% acceptable translations from the test-set. Unknown words mechanism improved translation quality by 3.56 points in human evaluation. This research also proposed the way to auto generate the explanation of each concept using the semantic backgrounds provided by UNL Ontology. These explanations are useful for improving translation quality of unknown words.

Acknowledgments

At first, I would like to thank Professor Nishino Tetsuro of The University of Electro-Communications, Tokyo for supervising my PhD thesis work to completion. I am really grateful to Professor Nishino Tetsuro for providing lot of opportunity and freedom in my research. His kind support and guidance allowed me to explore the vast research area of Machine Translation.

I want to give special thanks to my thesis committee members: Professor Takahashi Haruhisa, Professor Akira Utsumi, Professor Keiki TAKADAMA and Professor Maki Sakamoto. Their valuable suggestions and kind supports helped me to improve this thesis.

I am especially grateful to Dr. Yamada Setsuo for supporting in every step of my research in Machine Translation. Moreover he taught me reading and writing the technical papers with clear understanding. With his kind care and nurture now I feel confident to present my research in global platform.

I am also grateful to Dr. Hiroshi Uchida of UNDL Foundation, United Nations University for giving me the great opportunity to understand about his pioneering research in Machine Translation.

I am thankful to Dr. Shigeru Tanaka and Dr. Goto Takaaki for helping me with their expertise during my research by providing valuable suggestions.

I would like to thank every member of Nishino lab and my friends for giving me moral supports. I also want to thank my family members for providing their unconditional open-hearted love and support. I cannot show my gratitude using words to my parents. My mother Anwara Chowdhury is my inspiration and motivates me in each and every moment of my life, and, my father Md. Abdus Salam Khan gave me the vision for Machine Translation since my childhood.

TABLE OF CONTENTS

Chapter 1

Introduction

From the very beginning of computer history people dreamed that computer will able to translate human languages. Many researchers even declared their vision, that someday computer will replace human interpreters. IBM was the pioneer in this field and they developed the first working system during 1950s. This area of research is called as Machine Translation (MT). But still in this twenty first century, with the gigahertz processors and gigabyte memories, machines still fall flat when they try to translate languages. The latest programs provide some related results but only works properly in limited domains. For computer translating like human is still a very challenging task. With many novel research approaches computer achieved a level for translating languages in some extend. In fact MT was the hottest topic among Artificial Intelligence researchers in Computer Science discipline. However, Machine Translation is an interdisciplinary research area where all the experts from different areas has to come together to realize the dream. These experts are from Computer Science, Artificial Intelligence, Computational Linguistics, Linguistics, Cognitive Science, Information Processing, Software Engineering, and even Mathematics.

To develop efficient machine translation is very important but it is really expensive as it requires a huge amount of time and resources. In all languages there are many words that may have multiple meanings and also some sentence may have multiple grammar structure to express the same meaning, it is a great challenge to do the right semantic analysis. But it is very important to have machine translation which can compute all possible outputs in reasonable time and able to choose the best option.

In present there are many ways of machine translation. Many researchers came up with different approaches. But still it is not possible to get the finest possible result. I want to use the example-based machine translation, to get all possible outputs. For achieving this I have to plan to prepare a dictionary with morphological analysis and a Parallel Corpus. Then from semantic analysis it may possible to choose the best desired output.

In this thesis we considered Example-Based Machine Translation (EBMT) for low resource language, like Bengali. Due to the lack of parallel corpus, it has high probability of handling unknown words which cause low-coverage issues. In this research, we propose an EBMT for low resource language using chunk-string templates (CSTs). CSTs consist of a chunk in source-language, a string in target-language, and word alignment information. CSTs are prepared automatically from aligned parallel corpus and WordNet by using English chunker. In our experiment CSTs improved the wide-coverage by 57 points and quality by 48.81 points in human evaluation. Currently 64.29% of the test-set translations by the system were acceptable. The combined solutions of CSTs and unknown words generated 67.85% acceptable translations from the test-set. Unknown words mechanism improved translation quality by 3.56 points in human evaluation..

In chapter 2 we discussed about Machine Translation in brief. Here the beginners can get a brief idea about what is Machine Translation, A Brief History of Machine Translation, Machine Translation Approaches etc.

In chapter 3, we discussed about Example-Based Machine Translation (EBMT). To give a brief idea about EBMT, we discussed about the general EBMT Architecture. For readers understanding we explained why we should choose Example-based Machine Translation. However as always EBMT is not the best choice, we dicussed about the difficulties of EBMT afterwards. Initial requirements for EBMT are discussed as well.

In chapter 4, we proposed our EBMT architecture for English-Bengali language pair.

In chapter 5, we discussed about the development methodology of English-Bengali parallel corpus.

In chapter 6, we discussed about our proposal of using Chunk-String Templates (CSTs) in EBMT.

In Chapter 7, we discussed about our solution for translating unknown words.

In Chapter 8, we discuss our experiment result from the implementation of the proposed EBMT system

In Chapter 9, we discussed about expansion of our research using UNL ontology.

CHAPTER 2.

MACHINE TRANSLATION IN BRIEF

2.1 What is Machine Translation?

Machine Translation is the process of translating text units of source language into a target language by using computers. The term Machine Translation can be defined as "translation from one natural language (source language (SL)) to another language (target language (TL)) using computerized systems, with or without human assistance" Hutchins and Somers pg. 3 [45]

The ideal MT system would support speech-to-speech-translation. However in reality it is divided into three parts:

Part1: Speech-to-Text
Speech-to-Text only works with sound or speech data. For example they only deal with transcribing Japanese speech in text format.

Part2: Machine Translation
This is the main part where computer actually translate the language. For Example Computer translate that Japanese text into English. In this thesis we only considered about this part.

Part3: Text to Speech

In this part the computer produce sound from the translated texts produced in earlier step. For Example Computer Produce sound from that translate English.

In general researchers become expert in any one of these areas. In fact these three are completely different research area as they handle different issues. Each area has different conferences or journals. Combining the experts from these areas the ideal MT systems were developed.

Machine Translation process can be explained using Figure 2.1. Input of Machine Translation is texts or sentences in source language, for example in English. And the output of the Machine Translation is the translation of the input texts or sentences in target language, for example in Bengali.

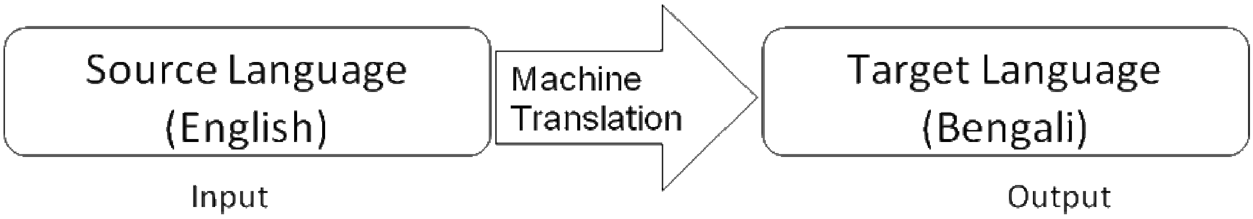


Figure 2.1: Machine Translation Process

Machine translation is referred by the abbreviation MT. Reader should not get confused with computer-aided translation, machine-aided human translation MAHT and interactive translation. As these fields are sub-field of computational linguistics that investigates the use of computer software to translate text or speech from one natural language to another.

Simple MT performs simple substitution of words in one natural language for words in another, but that alone usually cannot produce a good translation of a text, because recognition of whole phrases and their closest counterparts in the target language is needed. Solving this problem with corpus and statistical techniques is a rapidly growing field that is leading to better translations, handling differences in linguistic typology, translation of idioms, and the isolation of anomalies

2.2 A Brief History of Machine Translation

If we define machine just as a tool not only electronic device then machine translation idea can be traced back to 1629. René Descartes, a French philosopher first proposed a universal language, with equivalent ideas in different tongues sharing one symbol. In some extent we can

IBM[1] was the pioneer in developing machine translation devices. IBM founder Thomas Watson Sr. experienced with language barriers while working in International Chamber of Commerce. In 1927, under his leadership, IBM developed the very first translation system based on the Filene-Finlay simultaneous translator. It allowed users to listen the professional translator's speeches in real time. In 1931, the IBM-Filene-Finlay translator was installed at the League of Nations in Geneva. There, users could listen the pre-translated speeches simultaneously, while interpreters took notes. Then interpreter would give the speech in his own language. During the Nuremberg war crime trials of 1946 this system required modification for true simultaneous interpretation. Here speakers speak slowly so that all interpreters could speak along with them.

During 1950s, IBM developed an English-Russian translator using the IBM 701 Electronic Data Processing Machine. It incorporated logic algorithms that made grammatical and semantic "decisions" to human translators. The Georgetown experiment (1954) involved fully automatic translation of over sixty Russian sentences into English. The experiment was a great success and managed to get good funding for machine-translation research. Moreover they claimed that within three to five years, machine translation would be a solved problem. Obviously even after 60 years machine translation is an unsolved problem. This is the well known story which researchers refer as "3 to 5 years" joke in machine translation. Even now some inexperienced researchers without knowing this joke would claim that within "3 to 5 years" their system can solve the language barrier. That is why every researcher should study the history of their research area to make new claims.

ALPAC report (1966) found that the ten-year-long research had failed to fulfill expectations. As a result funding was greatly reduced. Beginning in the late 1980s, as computational power increased and became less expensive, more interest was shown in statistical models for machine translation.

2.3 Machine Translation Approaches

In present there are many ways of machine translation. Many researchers came up with different approaches. But still it is not possible to get the finest possible result. I want to use the example-based machine translation, to get all possible outputs. For achieving this I have to plan to prepare a dictionary with morphological analysis and a Parallel Corpus. Then from semantic analysis it may

[1] http://www.ibm.com/ibm100/us/en/icons/translation/

possible to choose the best desired output.

Machine translation approaches can be divided in two generations direct systems and indirect systems. First generation systems are known as direct systems referred as Rule Based MT (RBMT). In RBMT, translation is done word by word or phrase by phrase. If it is done word by word it is called Dictionary Based MT.

Knowledge Based MT (KBMT) approach was also popular at the beginning. In such systems very minimal linguistic analysis of input text is conducted [45]. This architecture is still being extensively used in commercial MT systems. The main idea behind direct systems is to analyze the input text to the extent that some transformational rules can be applied. This analysis could be parts of speech of words or some phrasal level information. Then using a bilingual dictionary, source language words are replaced with target language words and some rearrangement rules are used to modify the word order according to the target language [21].

KBMT or RBMT are very robust because it does not fail on any erroneous or ungrammatical input. Since the analysis level is very shallow and the system contains very limited grammatical information, it hardly considers anything ungrammatical. In the worst case if the rule does not apply to the input, the input is passed on without any alteration as output. This kind of system is hard to extend because all the rules are written in one direction and are language specific. To make another language pair work, all the rules have to be re-written. Since the system does not perform very deep analysis, its time complexity is low. These systems work very well for closely related languages but are not suitable for modeling languages with diverse syntactic nature. Since the system does not explicitly contain the grammatical rules of the target language, there is a chance that the output will not be grammatical but it will be similar to the target language (Arnold et al. 1993)

SYSTRAN is one of the most well-known direct systems [45]. Indirect systems can be further divided into Interlingua and Transfer based systems.

Owing to the fact that linguistic information helps an MT system to produce better quality target language translation, with the advance of computing technology, MT researchers started to develop methods to capture and process the linguistics of sentences. This was when the era of second generation MT systems started. Second generation machine translation systems are called indirect systems. In such systems the source language structure is analyzed and text is transformed into a logical form. The target language translation is then generated from the logical form of the text

[45]. The transition from direct systems to indirect systems is illustrated as Machine Translation Pyramid in Figure 2.2, taken from pg. 107 of [45]. Using this Machine Translation Pyramid we can actually explain any Machine Translation approach.

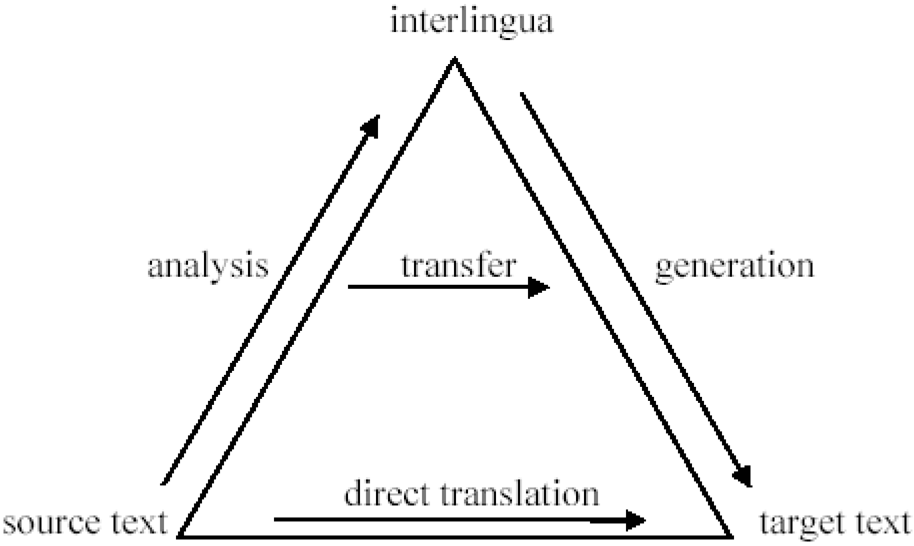


Figure 2.2: Machine Translation Pyramid

In the transfer method, the source language is analyzed to an abstract level. Then, through a transfer module, this abstract form is converted to the corresponding abstract form in the target language through which the target translation text is generated. The module '*SL Analysis*' captures the required linguistic information about the source language sentences to aid the translation. '*SL to TL Transfer*' module transfers the representation generated by '*SL Analysis*' to a target language representation. The module '*TL Generation*' generates the translation text using this logical representation. Such a system requires independent grammars for the source and target languages. Moreover it requires a comparative grammar or transfer roles to relate source structures to target structures. Since the system assumes full grammatical knowledge it does not allow ungrammatical sentences to be parsed, thus reducing the output of the system. This kind of system is easy to extend because to add a new language, grammar and transfer rules for the new language need to be written but the grammar of the other language is reusable. Such systems are theoretically reversible. The same grammars can be used in the reversed system. Practically there are problems in reversing the system because some transfer rules which are correct in one direction may not be correct in the other direction. The system has the explicit grammar of the target language, which ensures grammatical output [21]. Examples of transfer systems include ARIANE (Vauquois and Boitet 1985), SUSY (Maas 1987), MU (the Japanese National Project) (Nagao et al. 1986), METAL (Slocum et al. 1987; Bennett and Slocum 1988), TAUM-AVIATION (Isabelle 1987), ETAP-2

(Apresian et al. 1992), LMT (McCord 1989), EUROTRA (Arnold 1986; Arnold and des Tombe 1987; Copeland et al. 1991a,b), CAT-2 (Sharp 1988), MIMO (Arnold and Sadler 1990), MIMO-2 (van Noord et al. 1990) and ELU (Estival et al. 1990).

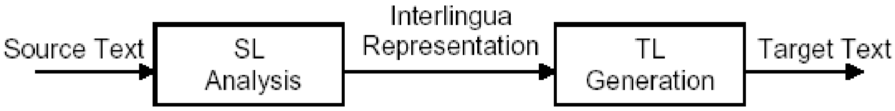


Figure 2.2a: Interlingua Based System

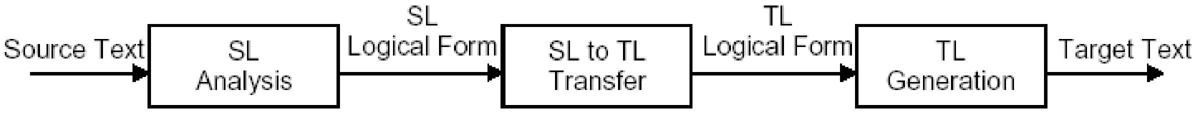


Figure 2.2b: Transfer Based System

The Interlingua approach involves the use of an intermediate language (i.e. an Interlingua) for the transfer, with the source language text translated to the Interlingua and the Interlingua translated to the target language text. As suggested by [45], an Interlingua is an intermediate 'meaning' representation and this representation:

"*includes all information necessary for the generation of the target text without 'looking back' to the original text. The representation is thus a projection from the source text and at the same time acts as the basis for the generation of the target text; it is an abstract representation of the target text as well as a representation of the source text.*" Hutchins and Somers p. 73 [45]

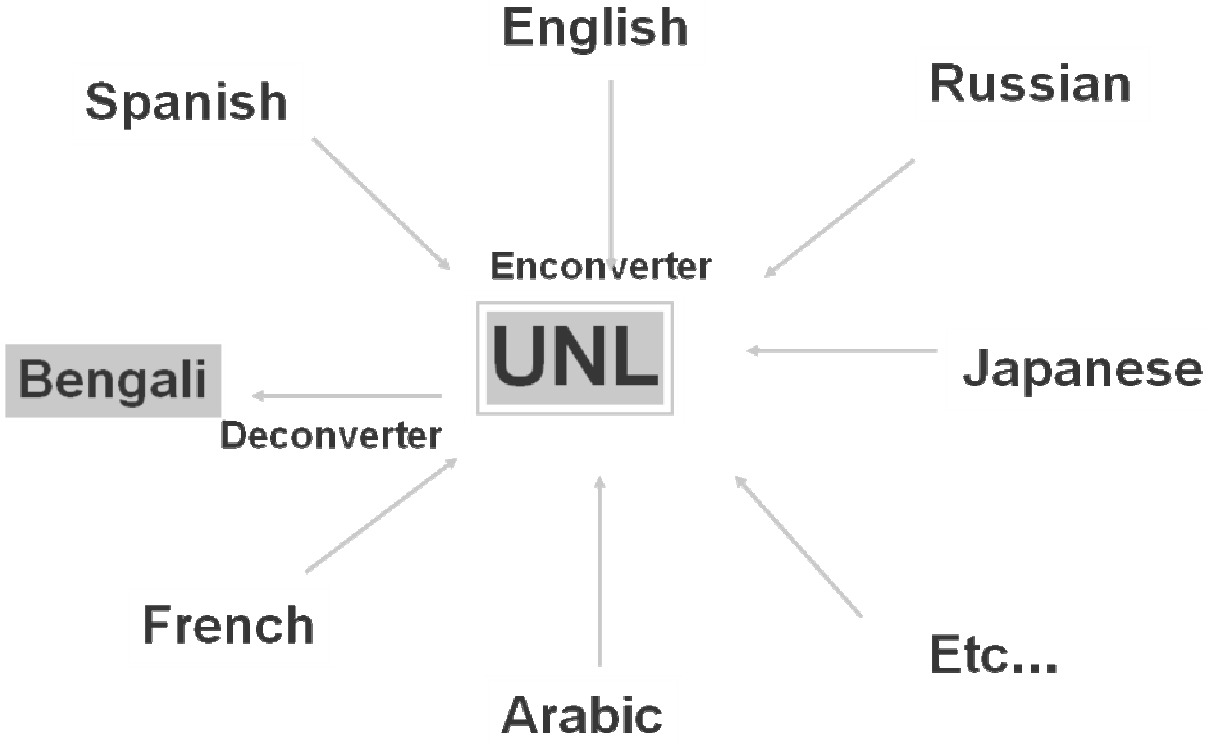


Figure 2.3: Adding Bengali to UNL

Interlingua appears to be an attractive approach for machine translation due to several reasons. Firstly, from a theoretical point of view it is very interesting to establish a representation which is independent of language. Secondly, Interlingua systems are more easily extendable because only analysis and generation modules are required to add a new language and no language specific transfer information is needed. But it is difficult to define such a language independent representation even for closely related languages [21].

An attempt to define an Interlingua to represent the language in the form of a semantic relation is The Universal Networking Language (UNL) project. This project was initiated by the University of United Nations based in Tokyo in 1996.

Figure 2.3 shows that if we develop Bengali DeConverter for UNL language it can translate from other languages such as English, Spanish etc. However UNL is rule based approach which is time consuming for writing the rules.

An utterance is represented as a hyper-graph in UNL. Normal nodes in the graph bear Universal Words (UWs) with semantic attributes and arcs bear semantic relations (deep cases, such as agt, obj, goal, etc.). UNL representation is being built in many languages including Arabic, Chinese, French, German, Hindi, Indonesian, Italian, Japanese, Mongolian, Portuguese, Russian, and Spanish. Some other Interlingua systems are Rosetta by Landsbergen 1987, KBMT by Goodman and Nirenburg [21].

There are new emerging approaches to MT known as the empirical approaches. They apply statistical or pattern matching techniques for MT. These techniques are called empirical since the knowledge for translation is derived empirically by examining text instead of linguistic rules. There are two such approaches, the ‘example’ or ‘analogy’ based approach, and the ‘statistical’ approach [21].

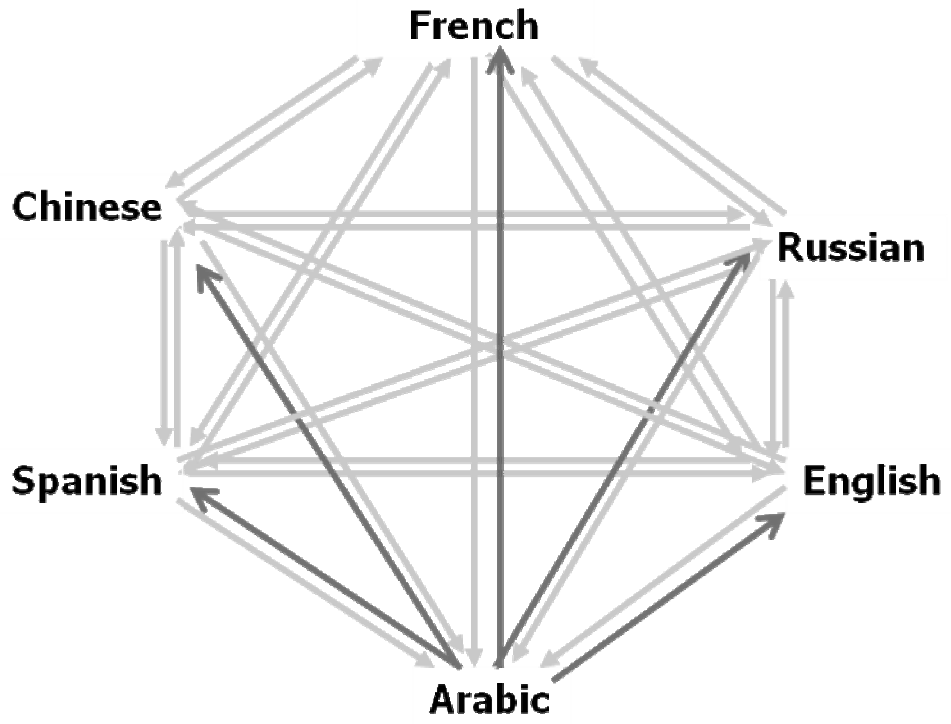


Figure 2.4: Transfer approaches require transfer system for each language pair.

EBMT and SMT are transfer approaches which require transfer system for each language pair. For example Figure 2.4 shows 5 * 6 = 30 transfer based systems requred for 6 United Nations official languages.

In the ‘example-based’ approach, translation is done by matching the given text with stored example translations. The basic idea is to collect a bilingual corpus of translation pairs and then use a best match algorithm to find the closest example to the source phrase to be translated. This gives a translation template, which can then be filled in by a word for word translation. A limitation of this technique is that it requires a large bilingual aligned corpus. But these examples can also be built incrementally, increasing the quality of translation. Such systems are efficient because they need not to go through complex grammars to analyze the text, but if many examples match the input text then finding the best match can be a complex task. A pure example based system will include no linguistic knowledge but addition of some linguistic knowledge can improve the system by increasing its capability of dealing with more patterns concisely as one can specify categories instead of raw words [21].

The second approach, the 'statistical approach', uses probabilistic analysis in MT as the name suggests. This term sometimes refers to the use of probability based techniques in parts of the MT task like word sense disambiguation or structural disambiguation. The other use of this term refers to a pure statistical machine translation, which uses probabilistic models to determine the correct translation of input text. In this approach, two statistical models, namely a 'language model' and a 'translation model' are built. This technique has been successfully used in speech recognition. A language model provides probabilities of occurrence of the sentence in the language, P(S) and a translation model provides probability of a target sentence given source sentence, P(T/S). An N-gram model is used to build the language model. Language models for both source and target languages are built. The translation model is computed using a word-level aligned bilingual corpus. For details of the modeling process, refer to Brown et al. (1990). Using language model probabilities and conditional probabilities of the translation model, P(S/T) is computed using the following formula:

$$P(S/T) = \frac{P(S)P(T/S)}{P(T)}$$

This approach does not require explicit encoding of linguistic information. On the other hand, it is heavily dependent on the availability of good quality bilingual data in very large proportions, which is currently not available for most languages [21].

Context-Aware Machine Translatio incorporate contextual information into the Translation and Language Models applied during machine translation. In our research we propose to incorporate contextual information using ontology. Context refers to surrounding text of an expression (word, phrase, chunk or sentence). The idea is that context influences the way we understand the expression. Hence the norm not to cite people out of context.

Generic translation services are context independent, because of that a sentence has same translation wherever it occurs. However ambiguities of natural language can only be resolved with the contextual information. Moreover, contextual information helps users to understand unknown words.

Chapter 3

Background

3.0 Introduction

Bengali is the native language of around 230 million people worldwide, mostly from Bangladesh. According to "Human Development Report 2009"[2] of United Nations Development Program, the literacy rate of Bangladesh is 53.5%. So we can assume that around half of Bengali speaking people are monolingual. Since significant amount of the web contents are in English, it is important to have English to Bengali Machine Translation (MT) system. But English and Bengali form a distant language pair, which makes the development of MT system very challenging. Bengali is considered as low-resource language, due to the lack of language resources like electronic texts and parallel corpus. As a result, most of the commercial MT systems do not support Bengali language translation.

In present, there are several ways of Machine Translation such as Rule-Based Machine Translation (RBMT), Statistical Machine Translation (SMT) and Example-Based Machine Translation (EBMT) which includes chunk-based and template-based approaches.

RBMT require human made rules, which are very costly in terms of time and money, but still unable to translate general-domain texts. There are several attempts in building English-Bengali MT system. The first available free MT system from Bangladesh was Akkhor Bangla Software[3]. The second available online MT system was apertium based Anubadok[4]. These systems used Rule-Based approach and did not consider about improving translation coverage by handling unknown

[2] http://hdr.undp.org/en/reports/global/hdr2009/

[3] www.akkhorbangla.com

words, in low-resource scenario. Dasgupta et.al. (2004) proposed to use syntactic transfer. They converted CNF trees to normal parse trees and using a bilingual dictionary, generated output translation. This research did not consider translating unknown words.

SMT works well for close language pairs like English and French. It requires huge parallel corpus, but currently huge English-Bengali parallel corpus is not available. English to Bengali phrase-based statistical machine translation was reported by Islam et al. (2010). This system achieved low BLEU score due to small parallel corpus for English-Bengali.

EBMT is better choice for low-resource language, because we can easily add linguistic information into it. Comparing with SMT, we can expect that EBMT performs better with smaller parallel corpus. Moreover, EBMT can translate in good quality when it has good example match. However, it has low-coverage issues due to low parallel corpus. Naskar et al. (2006a), reported a phrasal EBMT for translating English to Bengali. They did not provide any evaluation of their EBMT. They did not clearly explain their translation generation, specially the word reorder mechanism. Saha et al. (2005) reported an EBMT for translating news headlines. Salam et al. (2009) proposed EBMT for English-Bengali using WordNet in limited manner.

Chunk parsing was first proposed by Abney (1991). Although EBMT using chunks as the translation unit is not new, it has not been explored widely for low-resource Bengali language yet. Kim et al. (2010) used syntactic chunks as translation units for improving insertion or deletion words between two distant languages. However this approach requires an example base with aligned chunks in both source and target language. In our example-base only source side contains chunks and target side contains corresponding translation string.

Template based approaches increased coverage and quality in previous EBMT. Moreover, Gangadharaiah et al. (2011) showed that templates can still be useful for EBMT with statistical decoders to obtain longer phrasal matches. Manually clustering the words can be a time consuming task. It would be less time consuming to use standard available resources such as WordNet for clustering. That is why in our system, we used <lexical filename> information for each English words, provided by WordNet-Online for clustering the proposed CST.

3.1 EBMT in Brief

Example-based Machine Translation (EBMT) makes use of past translation examples to generate the translation of a given input. In other words the EBMT approach to machine translation is often characterized by its use of a bilingual corpus with parallel texts as its main knowledge base, at run-

[4] anubadok.sourceforge.net

time. It is essentially a translation by analogy and can be viewed as an implementation of case-based reasoning approach of machine learning.

An EBMT system stores in its example base of translation examples between two languages, the source language and the target language. These examples are subsequently used as guidance for future translation tasks. In order to translate a new input sentence in Source Language (SL), similar SL sentence is retrieved from the example base, along with its translation in Target Language (TL). This example is then adapted suitably to generate a translation of the given input. Figure 1 shows the role of example in EBMT.

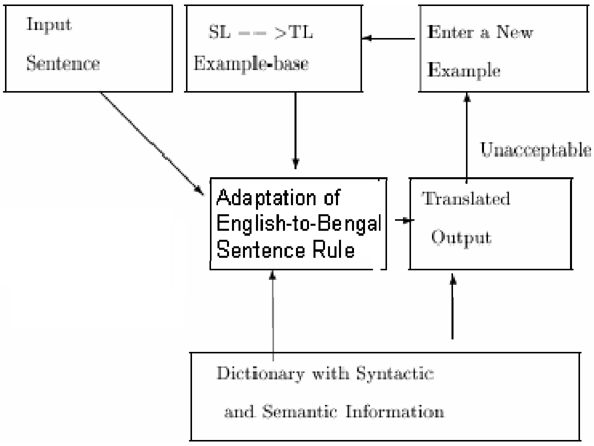


Figure 3.1: Role of Examples in EBMT

It may be observed that in today's world a lot of information is being generated. However, since most of this information is in English, it remains out of reach of people at large for which English is not the language of communication. As a consequence, an increasing demand for developing machine translation from English to Bengali is being felt very strongly.

However, development of MT systems typically demands availability of a large volume of computational resources, which is currently not available for Bengali [5]. Moreover, generating such a large volume of computational resources (which may comprise an extensive rule base, a large volume of parallel corpora etc.) is not an easy task. EBMT scheme, on the other hand, is less demanding on computational resources making it more feasible to implement in respect of these languages.

Let's see what wikipedia says about EBMT:

"At the foundation of example-based machine translation is the idea of translation by analogy. When applied to the process of human translation, the idea that translation takes place by analogy is a rejection of the idea that people translate sentences by doing deep linguistic analysis. Instead it is founded on the belief that people translate firstly by decomposing a sentence into certain phrases, then by translating these phrases, and finally by properly composing these fragments into one long sentence. Phrasal translations are translated by analogy to previous translations. The principle of translation by analogy is encoded to example-based machine translation through the example translations that are used to train such a system."

Example of bilingual corpus

English	Bengali
How much is that **red umbrella**?	oi **lal Chata** tar dam koto?
How much is that **small camera**?	oi **choto kamera** tar dam koto?

EBMT requires bilingual parallel corpora for training purpose. Bilingual parallel corpora contain sentence pairs like the example shown in the above table. Sentence pairs contain sentences in one language with their translations into another. The particular example shows an example of a minimal pair, meaning that the sentences vary by just one element. These sentences make it simple to learn translations of subsentential units. For example, an example-based machine translation system would learn three units of translation:

1. *How much is that* **X** *?* corresponds to *oi **X** tar dam koto?*

2. *red umbrella* corresponds to *lal Chata*

3. *small camera* corresponds to *choto kamera*

These small units can help to produce real time translations for future input sentences. For example, if we have been trained using some text containing the sentences: "President Kennedy was shot dead during the parade." and "The convict escaped on July 15th.". We could translate the sentence: "The convict was shot dead during the parade.", by substituting the appropriate parts of the sentences. Other approaches to machine translation, including statistical machine translation, also use bilingual corpora to learn the process of translation.

Example-based machine translation was first suggested by Makoto Nagao in 1984.[31] It soon attracted the attention of scientists in the field of natural language processing.

EBMT is best suited for sub-language phenomena like phrasal verbs. Phrasal verbs have highly context-dependent meanings. Phrasal verbs are a commonly occurring feature in English and comprise a verb followed by an adverb and/or a preposition. The adverb/preposition(s) are termed as the particle to the verb. Phrasal verbs produce specialized context-specific meanings that may not be derived from the meaning of the constituents. There is almost always an ambiguity during word-to-word translation from source to the target language. As an example, let us consider the phrasal verb: put on and its Bengali meaning. It may be used in any of the following ways: Ram put on the lights. (Switched on) (Jalano). Ram put on a cap. (Wear) (pora). EBMT can be used to determine the context of the sentence.

3.2 General EBMT Architecture

Figure 2 shows the general EBMT Architecture. Here EBMT trained on processed parallel corpus and translate input sentences based on those translation resources.

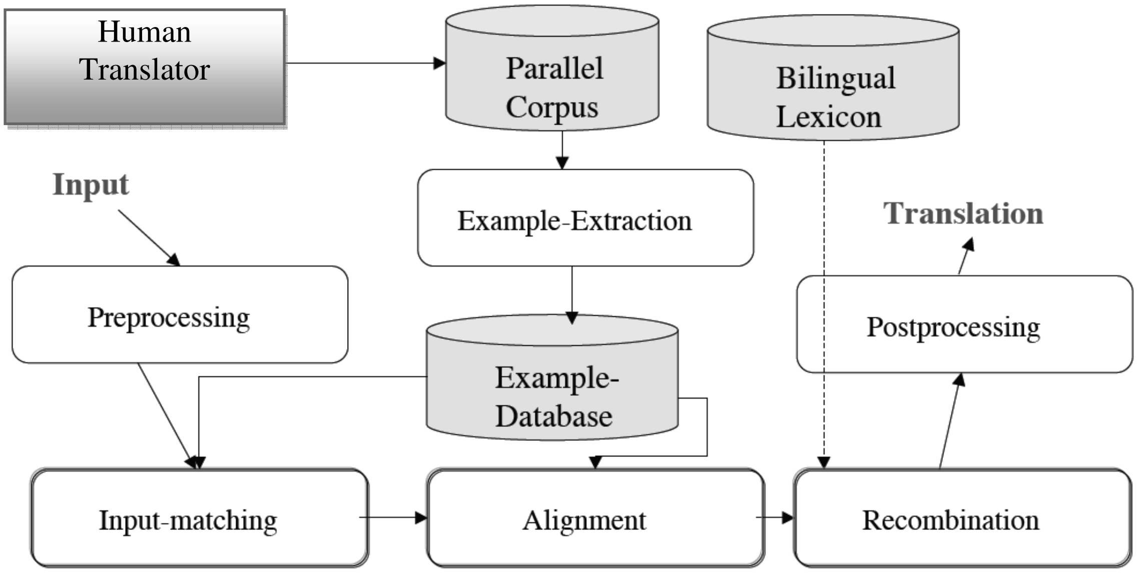


Figure 3.2: General EBMT Architecture

3.3 Why Example-based Machine Translation?

Development of MT systems typically demands availability of a large volume of computational resources, which is currently not available for Bengali or other low resource language. Moreover, generating such a large volume of computational resources (which may comprise an extensive rule base, a large volume of parallel corpora etc.) is not an easy task. EBMT scheme, on the other hand, is less demanding on computational resources making it more feasible to implement for low resource language like Bengali.

Example-based Machine Translation makes use of past translation examples to generate the translation of a given input. An EBMT system stores in its example base of translation examples between two languages, the source language and the target language. These examples are subsequently used as guidance for future translation tasks. In order to translate a new input sentence in SL, similar SL sentence is retrieved from the example base, along with its translation in TL. This example is then adapted suitably to generate a translation of the given input. It has been found that EBMT has several advantages in comparison with other MT paradigms (Sumita and Iida, 1991):

1. It can be upgraded easily by adding more examples to the example base;
2. It utilizes human translators' expertise, and adds a reliability factor to the translation;
3. It can be accelerated easily by indexing and parallel computing;
4. It is robust because of best-match reasoning.

Even other researchers, like Somers or Kit, have considered EBMT to be one major and effective approach among different MT paradigms, primarily because it exploits the linguistic knowledge stored in an aligned text in a more efficient way. We apprehend from the above observation that for development of MT systems from English to Bengali, EBMT should be one of the preferred approaches. This is because a significant volume of parallel corpus is available between English and Bengali in the form of Newsletters, Bi-lingual websites, government notices, translation books, advertisement material etc. Although all data is generally not available in electronic form yet, converting them into machine readable form is much easier than formulating explicit translation rules as required by an EBMT system.

3.4 Difficulties of Example-based Machine Translation

Initially for small parallel corpus EBMT perform better than SMT. But for huge parallel corpus SMT performs better. However in both cases we can not use the system for general purpose translation unless we have huge parallel corpus. So it becomes improvable by increasing Knowledge Base. Because to match Example-base become very difficult with many candidates. SMT has open source tools like moses. But currently there is no open source tools available for EBMT.

3.5 Initial Requirement for EBMT

For General EBMT the only requirement is parallel corpus. However if you want statistical generation you also need to prepare Language Model. Based on EBMT model you may need bilingual Dictionary (English to Bengali) with morphological analysis information.

We also have to find suitable existing solutions for the following aspects:

a) Development of efficient retrieval and adaptation scheme: Efficient adaptation of past examples is a major aspect of an EBMT system. There are many adaptation schemes available for an EBMT system. Even an efficient similarity measurement scheme and a quite large example base cannot guarantee an exact match for a given input sentence. As a consequence, there is a need for an efficient and systematic adaptation scheme for modifying a retrieved example, and thereby generating the required translation.

b) Study of divergence for your language translation (eg. English to Bengali), and how translation divergence can be effectively handled within an EBMT framework.

c) How to handle complex sentences - which are in general considered to be difficult to deal with in an MT system.

3.5 Related Research

In this section, I will write about my understanding about some references. For the publication information please see in the reference section.

Chunk parsing was first proposed by Abney [31]. Although EBMT using chunks as the translation unit is not new, it has not been explored widely for low-resource Bengali language yet. Kim et al. used syntactic chunks as translation units for improving insertion or deletion words between two distant languages [35]. However this approach requires an example base with aligned chunks in both source and target language. In our example-base only source side contains chunks and target side contains corresponding translation string.

Template based approaches increased coverage and quality in previous EBMT. Moreover, Gangadharaiah et al. showed that templates can still be useful for EBMT with statistical decoders to obtain longer phrasal matches [40]. Manually clustering the words can be a time consuming task. It would be less time consuming to use standard available resources such as WordNet for clustering. That is why in our system, we used <lexical filename> information for each English words, provided by WordNet-Online for clustering the proposed CST.

Dasgupta et.al. proposed to use syntactic transfer. They converted CNF trees to normal parse trees and using a bilingual dictionary, generated output translation [41]. This research did not consider translating unknown words.

Naskar et al. reported a phrasal EBMT for translating English to Bengali [42]. They did not provide any evaluation of their EBMT. They did not clearly explain their translation generation, specially the word reorder mechanism.
Saha et al. reported an EBMT for translating news headlines [32]. Their works showed that EBMT can be a good approach for Bengali language. Their approach only considered about news headlines.
English to Bengali phrase-based statistical machine translation was reported by Islam et al. [39]. This system achieved low BLEU score due to small parallel corpus for English-Bengali.
Salam et al. proposed EBMT for English-Bengali using WordNet in limited manner. [36]

Chapter 4

Proposed EBMT Architecture

In today's world a lot of information is being generated around the world in various fields. However, since most of this information is in English, it remains out of reach of Bangladeshi people for which English is not the language of communication. As a consequence, an increasing demand for developing machine translation systems from English to Bengali is being felt very strongly. The primary goal of the proposed EBMT architecture considered English to Bengali MT.

In this thesis, we propose an EBMT for low re-source language, using chunk-string templates (CSTs) and translating unknown words. CSTs consist of a chunk in the source language (English), a string in the target language (Bengali), and the word alignment information between them. CSTs are generated from the aligned parallel corpus and WordNet, by using English chunker. WordNet (Miller 2005) is a large lexical database of English, where nouns, verbs, adjectives and adverbs are grouped into clusters using <lexical filename> information. For clustering CSTs, we used <lexical filename> information for each words, provided by WordNet-Online .

To translate unknown words we used WordNet hierarchy of hypernym tree and an English-Bengali dictionary. At first the system finds the set of hypernyms words and degree of distance from the English WordNet. Then the system tries to find the translation of hypernym words from the dictionary according to the degree of distance order. When no dictionary entry found from the hypernym tree, it transliterates the word.

4.1 EBMT Architecture

The Figure 4.1 shows the proposed EBMT architecture. The dotted rectangles identified the main contribution area of this research. During the translation process, at first, the input sentence is parsed into chunks using OpenNLP Chunker.

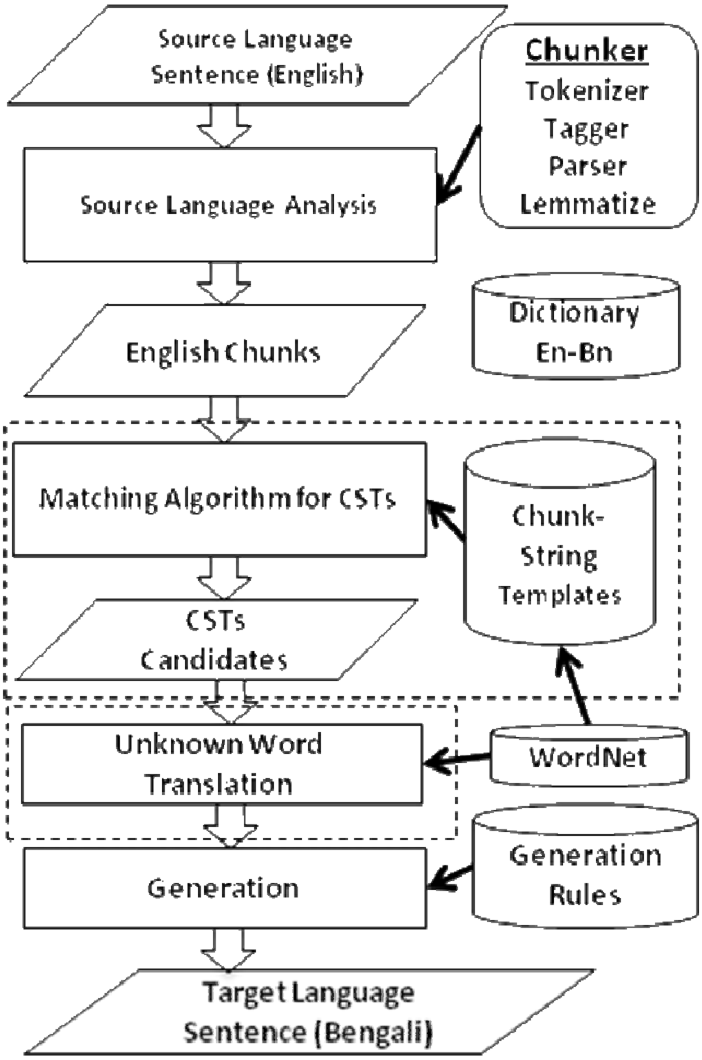


Figure 4.1: Proposed EBMT Architecture

The output of Source Language Analysis step is the English chunks. Then the chunks are matched with the example-base using the Matching algorithm as described in section 6.2. This process provides the CSTs candidates from the example-base and it also identify the unknown words in CSTs. In unknown word translation step, using our proposed mechanism in chapter 7, the system find translation candidates for the identified unknown words from WordNet. Then in Generation process WordNet helps to translate determiners and prepositions correctly to improve MT quality (Salam et al. 2009). Finally using the generation rules we output the target-language strings. Based on the EBMT system architecture in Figure 1, we built an English-to-Bengali EBMT system.

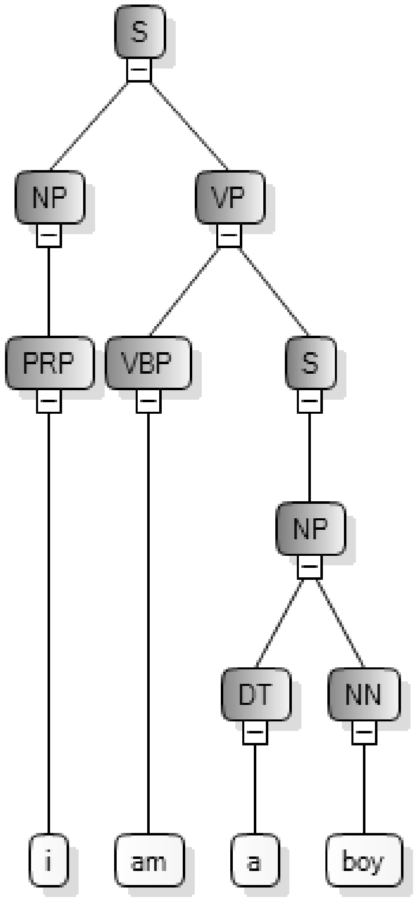
4.1 Tagging and Parsing

Tagging, is the process of marking up the words in a text as corresponding to a particular part of speech, based on both its definition, as well as its context—i.e., relationship with adjacent and related words in a phrase, sentence, or paragraph. Eg. I do-> I/PRP do/VBP

Parsing, is the process of analyzing a sequence of tokens to determine grammatical structure with respect to a given formal grammar. We used the tag set of Table4.1 for tagging the English sentence. Eg. I am a boy->(S (NP (PRP I)) (VP (VBP am) (NP (DT a) (NN boy))))

Table4.1: Tag set used for English to Bengali EBMT

Level 1	**Level 2**	**Tag**
Noun	Common	NN
	Proper	NNP
	Compound Common Noun	NNC
	Compound Proper Noun	NNPC
	Verb Root	NNV
	Temporal	NNT
	Question Temporal	QNT
	Locative	NNL
	Question Locative	QNL
Pronoun	Personal Pronoun	PRP
	Question Pronoun	QPR
Adjective	Simple	JJ
	Verb Root	JJV
	Question Adjective	QJJ
Vocatives	Vocatives	VOC
Verb	Main Finite Verb	VB
	Nonfinite Nominal	VBM
	Nonfinite Conditional	VBC
	Nonfinite Perfective	VBT
	Nonfinite	VBF
	Past tense	VBD
	Gerund/present participle	VBG
	Past participle	VBN
	Non-3rd ps. sing. Present	VBP
	3rd ps. sing. Present	VBZ
	Existential	VBE
Adverb	Adverb	RB
	Question Adverb	QRB
Conjunction	Co-ordinating	CC
	Compound Co-ordinating	CCC



	Suspicion	CN
	Eternal Joining	CET
	Subordinating	CS
	Compound Subordinating	CSC
Numbers	Cardinal Numbers	CD
Adposition	Adposition	ON
Interjection	Interjection	UH
Particle	Particle	RP
	Question Particle	QRP
Determiner	Common	DT
	Singular	DTS
	Question Determiner	QDT
Quantifier	Quantifier	QF
Foreign Word	Foreign Word	FW
Symbol	Symbol	SYM
List Item Marker	List Item Marker	LS
Suffixes	Adpositional	SFON
	Accusative	SFAC
	Possessive	SF$
Punctuation Marks & Others	Sentence Final Punctuation	.
	Comma	,
	Colon, Semi-colon	:
	Dash, Double-Dash	-
	Left ParenIndependent Study	(
	Right ParenIndependent Study	)
	Opening Left Quote	LQ
	Closing Right Quote	RQ
	Preposition/subordinate conjunction	IN

	Adjective, superlative	JJS
	Adjective, comparative	JJR
	Modal	MD
	Proper noun, plural	NNPS
	Noun, plural	NNS
	Predeterminer	PDT
	Possessive ending	POS
	Possessive pronoun	PRP$
	Adverb, comparative	RBR
	Adverb, superlative	RBS
	to	TO
	wh-determiner	WDT
	wh-pronoun	WP
	Possessive wh-pronoun	WP$
	wh-adverb	WRB
	Left open double quote	``
	Comma	,
	Right close double quote	'
	Sentence-final punctuation	.
	Colon, semi-colon	:
	Dollar sign	$
	Pound sign	#
	Left parenthesis	-LRB-
	Right parenthesis	-RRB-

4.2 Handle Complex Sentence Using Sub-Sentential EBMT:

Handling complex sentence in general considered to be difficult to deal with in an MT system. Since exact sentence matches only occur in special domains, we want to extend this to sub-sentence matches. For this we need to:

- Find the most similar example (involves segmenting by preparing chunks)
- Alter source side to match current input.

Similarity requires a “distance metric” in the source language (English). This can be closeness:

- of the lexical items in a hierarchy of terms/ concepts from ontology
- of the sequence of syntactic categories and function words,

- of the two syntactic structures,
- or combinations of these.

For these issues, in this thesis we proposed to use CSTs as described in next section

4.3 Adapting Scheme to Match Example-base

Efficient adaptation of past examples is a major aspect of an EBMT system. There are many adaptation schemes available for an EBMT system. Even an efficient similarity measurement scheme and a quite large example base cannot guarantee an exact match for a given input sentence. As a consequence, there is a need for an efficient and systematic adaptation scheme for modifying a retrieved example, and thereby generating the required translation. In chapter 5 we discuss details about our proposed adaptation scheme. In Table1 we gave a sample knowledge base of the English to Bengali EBMT System. During translation our adapting scheme chooses the best example for the source sentence.

4.4 Match the Example-base

Study of divergence for English to Bengali translation is also required. Translation divergence can be effectively handled within an EBMT framework. As in earlier step we have the sample rule and the parsed sentence. Now we can easily translate the sentence by matching the rule.

Study of divergence for English to Bengali translation is also required. Translation divergence can be effectively handled within an EBMT framework. As in earlier step we have the sample rule and the parsed sentence. Now we can easily translate the sentence by matching the rule.

- I am a boy > ami ekti chele
- I am a man > ami ekjon manus

In these two examples “a” has different meaning in Bengali “jon” and “ti”. Here we can see that it has same Example-base but different translation. Depending on the quality of the word we are choosing the actual meaning. Using WordNet we are determining that word sense. This technique dramatically improves the quality of EBMT.

For all birds plural we can use -kul

- Birds are flying > pakhikul akashe urchhe
- Parrots are flying> totapakhikul akashe urchhe

But for trees we have to use -raji

- Trees give us food> brikkhoraji amader khaddo dey

From the above example we see that in Bengali based on Noun quality different pos-fix used. Using WordNet and Table4.1 we can easily identify the ambiguity and translate correctly.

Chapter 5

English-Bengali Parallel Corpus

In this chapter we proposed the development process for English-Bengali parallel corpus. Bengali has more than 230 million native speakers worldwide. But it is considered as a low resource language. So, building a parallel corpus for Bengali language gets high priority.

The need of parallel corpus for corpus based studies in Natural Language Processing is well established. It is important to have a balanced parallel corpus for Machine Translation, automatic lexical acquisition, information extraction or Second language teaching. However, pairing with Bengali there is no such parallel corpus available yet. Therefore, all the parallel corpus based studies in Natural Language Processing remained undiscovered for Bengali language.

In this chapter we described our corpus selection criteria in the context of low resource language like Bengali. We followed the recent suggestions for corpus development, which is also useful for other low resource language like Bengali. We came up with the novel selection criteria: genre, style, mode, Domain, medium, time, author, writing level and target audience. We also identified the primary data sources and collection process which are available in both English and Bengali. The written texts was chosen from different domains of fine arts, philosophy, commerce, legal documents, natural sciences, social sciences, general/leisure, literature and others. As we have many electronic resources available online now, this chapter considered the online text sources with high priority in the design principal.

In background section we discussed about previous research works and the need of a balanced English-Bengali parallel corpus. In methodology section we described our selection criteria, domain balance issues and sample text collection sources. This guideline can be useful for the parallel corpus development of other low resource languages. In Additional Processing and Output, we discussed about the text processing issues, Tag Set and sample aligned sentences.

5.1 Background

A parallel corpus contains a source text and its translation into one or more target languages. The effectiveness of all corpus based studies in Natural Language Processing depends on the quality of parallel corpus. So it is very important to have a balanced parallel corpus for source and target language.

“Balanced corpus” refers to carefully selected and fully described body of natural language texts, which more or less represent the language. As it is a sample of a population it has the problem of sampling like any other division of science (McEnery and Wilson 2001). As language is an open set, it is never possible to contain all possible sentences in a corpus. We can always write a new sentence which was never written before. Therefore, it is very challenging task to build a balanced corpus. But it is very important to make the parallel corpus balanced. if the parallel corpus is not balanced then the statistical information produced from this will be skewed, which will provide unrepresentative word weights.

Frankenberg-Garcia, A. and Santos, D. (2003) introduced COMPARA, the Portuguese-English parallel corpus. They did not address the parallel corpus balancing issue in the corpus design principle. In our design methodology we considered the conventional corpus design issues such as balance and representativeness.

Chen et al. (1996) published the design methodology for SINICA Balanced Corpora for Chinese language with the size of around five million words. They also faced the problem of identify the sampling criteria for a balanced corpus. The corpus design principal had five attributes: source, mode, style, topic and genre. These attributes are promising and SINICA corpus is proven to be very useful. However for building parallel corpus these five attributes are not enough.

Dash, Niladri Sekhar and Chaudhuri, B.B. (2001) published the process of generation of a Bengali text corpus to explores the linguistic features noted within the text corpus. Although it was a nice attempt to understand formal and functional aspects of Bengali characters it was done under many limitations. They collected data from various text documents published within 1980 and 1995. Their main purpose was to understand the form and function of the characters, trace their behavioral peculiarities. However this corpus mainly considered the texts produced in India and it ignored the Bengali language publications from Bangladesh. There are some differences between the Bengali used in India and Bangladesh. This corpus was not developed in Standard Bengali Unicode format.

The main goal of Yeasir Arafat et al. (2006) was to built Prothom-alo Bengali monolingual newschapter corpus. They proposed the converting methodology of ASCII texts to Unicode format as well as producing the Bengali corpus with annotation.

5.2 Selection Criteria

Corpus selection criteria is very important to maintain the balancing factors. Our selection criteria is based on Chen et al. (1996). As we have to consider two different languages with different characteristics, the target audience and text translation or produce time need to be considered. The source and target text can be from English to Bengali or vise versa. The translation quality and each sentence writing level need to be considered as well. Considering this issues we added four new attributes' time, author, writing level and target audience. As language changes over time we need to have the published time of each text. Considering SINICA corpus design methodology, for parallel corpus five attributes are not enough. Having more parameters will enable us to achieve the ideal parallel corpus by adjusting the attributes. Specifically we added four more attributes: time, author, writing level and target audience. After surveying other language parallel corpus we came up with the novel selection criteria and the balancing factors of the corpus. To build the balanced English-Bengali parallel corpus we depend on nine independent selection criteria: genre, style, mode, domain, medium, time, author, writing level and target audience. We propose to include texts in our corpus following these selection criteria given in Table5.1.

In parallel corpus we have to consider two different languages with different characteristics. Here the target audience and text publication time need to be considered. The source and target text can be from English to Bengali or vise versa. The translation quality and each sentence writing level need to be considered as well. Considering this issues we added four new attributes' time, author, writing level and tar-get audience. As language changes over time we need to have the published time of each text.

Time Criterion: The time criterion refers to the date of publication of a text. Our balanced English-Bengali language corpus should contain texts from post Bangladesh period only. As language changes over time we have decided to include the texts which were published or revised after the year 1971, when Bangladesh became Independent. Therefore, this corpus is specifically targeted for the Bengali language used in Bangladesh. For English language portion this corpus keeps the mark of British English and American English wherever possible.

Table 5.1: corpus selection criteria list

Genre	
written	reportage
	patents
	commentary
	advertisement
	email
	announcement
	fiction
	localization texts
	academic prose
	Blogs
	poetry
	analects
	technical manual
spoken	movie/video script
	conversation
	speech
	meeting minutes
Style	narration
	argumentation
	exposition
	description
Mode	written
	written-to-be-read
	written-to-be-spoken
	spoken
	spoken-to-be-written
Domain	legal documents
	natural sciences
	social sciences
	philosophy
	fine arts
	commerce
	general/leisure
	literature

	other
Medium	newschapter
	encyclopedia
	academic journal
	textbook
	reference book
	website
	general magazine
	thesis
	general book
	audio/visual media
	interactive speech
Time	Publish/Review Date
Author	name, age, gender, region, language, publisher
Writing Level	1- 10 (simple-literary) formal/informal
Target Audience	age group, gender region, occupation

5.2.1 Data Source and Collection Process

To serve specific domain purpose, sub corpora can be defined in data source. The Domain of a text indicates the kind of writing it contains. Table8.2 shows the current Domain balance percentage. These percentages are the approximation of data availability.

Table8.2: Domain Balance Percentage

Domain	**%-age**
fine arts	5%
Other	5%
philosophy	10
commerce	10%
legal documents	10%
natural sciences	10%
social sciences	10%
general/leisure	20%
literature	20%

Total	100%

The written texts was chosen from different domains of fine arts, philosophy, commerce, legal documents, natural sciences, social sciences, general/leisure, literature and others categories. In consultation with the Society of Authors, the Publishers Association and other interested parties we got the permissions clearance.

Literature: Many fictions are translated from English to Bengali. But those were adaptation of English fiction. In this section quotes translations are also available. People frequently use excerpts from popular books and quotes. There are translated documents from different languages is also available. As in Bengali we accepted many foreign words these translations are also important.

Academic prose: Although it is very rare, there are some translations of academic books and articles available from English to Bengali. Most of these books were originally written in English later translated to Bengali.

Text Books: English learning text books in Bengali language is a good and reliable source of parallel sentence with most basic grammar structures. Some of the text books has translation from English to Bengali. These resources were collected from National Curriculum & Textbook Board

Newschapter: This is one of the biggest sources of our parallel corpus. News agencies translate the international news from English to Bengali. And national news translated from Bengali to English. We collected several Newschapter corpus in recent time period

Media: Radio and TV news script are available in both English and Bengali. From media we also got some popular song lyrics translations. A large amount of unscripted informal conversation from different age, region and social classes in a demographically balanced way, together with spoken language collected in all kinds of different contexts, ranging from formal business or government meetings to radio shows and phone-ins.

Technical Manual: Many electronics products user manuals translation are available in both English and Bengali.

Movie Subtitle: Many Bengali movie/drama has the subtitles translated to English and vise versa.

Legal Documents: There are many legal documents available in English and Bengali. Bangladesh Government Constitution is one of such important source. As every official people use these documents frequently it has high priority. There are many patents official procedural documents, official forms, applications and agreements translation available.

5.3 Additional Processing and Output

Three ways of creating electronic versions of the texts were envisaged at the start of the Balanced Bengali Language Corpus project:

1. OCR: Using Optical character recognition we can get electronic texts from books. Hand editing will may still required, though, to correct scanning errors and insert textual mark-up.

2. Keyboarding : Right now Scanners are not efficient enough at recognizing Bengali texts typefaces, lower-quality typography, or handwriting. It would have taken longer to correct scanned output in such cases than it did for a trained typist simply to re-type the documents in full. So Typing is better for leaflets, hand-written items, and of course recorded speech.

3. Existing electronic texts : There are many texts already exist in electronic form in Bengali and English. Such as Wikipedia, Baglapedia, Newschapters, Magazines,

4. Encoding of Texts

Compilation procedure: After Collecting raw text, we are converting all ascii fonts text to Unicode.

5 Tagging

We used the Tag set used in Khan Md. Anwarus Salam et al. (2009) in the parallel corpus.

6 Corpus Size

To serve the general purpose, the target size of English-Bengali parallel corpus is more than 10 Million words. Because this size is effective for Statisitical Machine Translation.We are proposing this parallel corpus to be an open-ended corpus. So it will able to grow in any new direction based on our corpus users by following the basic design principles. So the responsibility of achieving more balance will depends on the users.

In this chapter we identified the primary data sources which are available in both English and Bengali. To serve specific domain purpose, sub corpora can be defined in data source. After surveying other language parallel corpus we came up with the novel selection criteria and the balancing factors of the parallel corpus. To build the balanced English-Bengali parallel corpus we depend on nine independent selection criteria: genre, style, mode, Domain, medium, time, author, writing level and target audience. This chapter explains the domain classifications and weight percentage for each domain based on statistical analysis. The written texts was chosen from different domains of fine arts, philosophy, commerce, legal documents, natural sciences, social sciences, general/leisure, literature and others categories. Tagset used in this corpus are given in refernce. This chapter briefly described the corpus data collecting and digitalization process. The character encoding for Bengali language used is Unicode. Using recent suggestions for alignment encoding, this chapter elaborated the idea in the context of low resource language like Bengali. In recent years web became very powerful and big source for building corpus. As we have many electronic resources available online now, this chapter considered the online text sources with high priority in the design principal. Because it is economic to build a new corpus using online resources as they are already in digital format. To serve the general purpose, the target size of English-Bengali parallel corpus is more than 1 Million words. As language changes over time we have decided to include the texts which were published or revised after 1971, when Bangladesh became Independent. As there are some differences between Bengali used in India and Bangladesh, we chose to include only the texts published from Bangladesh. Therefore, this corpus is specifically targeted for the Bengali language used in Bangladesh. For English language portion this corpus keeps the mark of British English and American English wherever possible. The English - Bengali parallel corpus will play a vital role in Bengali language processing. This corpus will be available to other researchers for use in language related research.

It is very important to have a balanced parallel corpus for machine translation and other corpus based studies. Currently there is no balanced corpus available for English-Bengali language. That is why we described the development process of the first balanced English-Bengali parallel corpus for future. Such a corpus need to consider about automating the process of corpus creation, so that the corpus will able to grow day by day by gathering new reliable resource from web. Our corpus building process is still on progress. After building the corpus, doing statistical analysis we can provide more statistical information. Then we can also evaluate our corpus.

Chapter 6.

Chunk-String Templates (CSTs)

In this research we proposed EBMT based on chunk-string templates (CST), which is especially useful for developing a MT system for high-resource in source language to low-resource in target language. CST consists of a chunk in the source language (English), a string in the target language (Bengali), and the word alignment information between them. From the English-Bengali aligned parallel corpus CSTs are generated automatically.

Table 6.1 shows sample word-aligned parallel corpus. Here the alignment information contains English position number for each Bengali word. For example, the first Bengali word "বিশ্বব্যাপী" is aligned with 11. That means "বিশ্বব্যাপী" is aligned with "worldwide", the 11th word in the English sentence. Although the last Bengali word "মাতৃভাষা" is aligned with 4, the word meaning includes "the native language". Therefore, the alignment information does not have 3rd and 5th words.

English	Bengali	Align

Bangla is the native language of
1 2 3 4 5 6
around 230 million people worldwide
7 8 9 10 11
	বিশ্বব্যাপী বাংলা হচ্ছে প্রায় ২৩০ মিলিয়ন মানুষ –এর মাতৃভাষা	11 1 2 7
		8 9 10 6
		4

Table 6.1: Example word-aligned parallel corpus

The example-base of our EBMT is stored as CST. CST consists of <c;s;t>, where c is a chunk in the source language (English), s is a string in the target language (Bengali), and t is the word alignment in-formation between them.

6.1 Generate CSTs

A chunk is a non-recursive syntactic segment which includes a head word with related feature words. In this paper OpenNLP has been used for chunking purpose. For example, “[NP a/DT number/NN]”, is a sample chunk. Here NP, DT, NN are parts of speech (POS) Tag defined in Penn Treebank tag set as: proper noun, determiners, singular or mass noun. The third brackets “[]” define the starting and ending of a complete chunk.

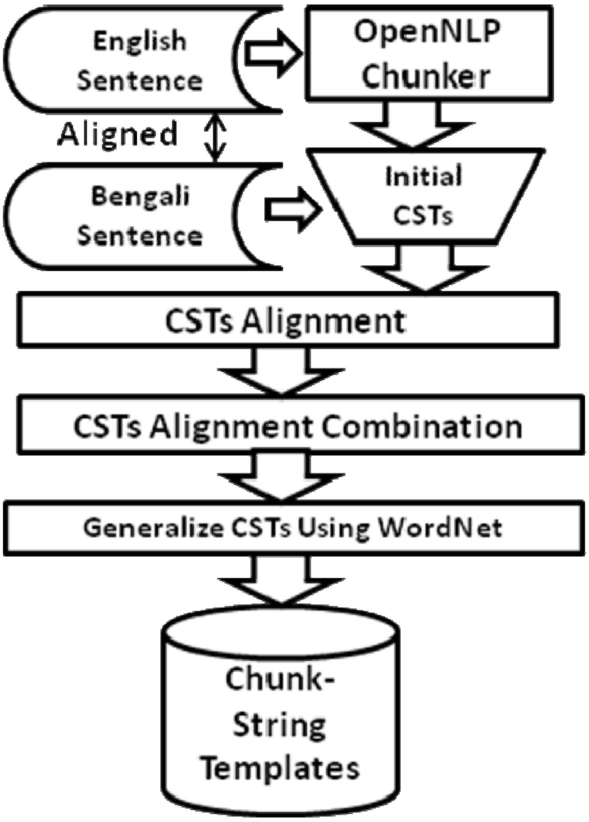


Figure 6.2: Steps of CSTs generation

Figure 6.2 shows the steps of CSTs generation. First the English chunks are auto generated from a given English sentence. Then initial CSTs are generated for each English chunks from the English-Bengali parallel corpus. Each CSTs alignment for all sentences are generated using the parallel corpus. After that the system generate combinations of CSTs. Finally the system produce CSTs by generalizing using WordNet to achieve wide-coverage.

6.1.1 OpenNLP Chunker

In the first step, using OpenNLP chunker, we prepare chunks of the English sentences from the word aligned English-Bengali parallel corpus.

Input of this step: *“Bangla is the native language of around 230 million people worldwide.”*

Output of this step: *“[NP Bangla/NNP] [VP is/VBZ] [NP the/DT native/JJ language/NN] [PP* *of/IN] [NP around/RB 230/CD million/CD people/NNS] [ADVP worldwide/RB] ./.”*

6.1.2 Initial CSTs

In this second step, initial CSTs are generated for each English chunks from the English-Bengali parallel corpus. Table 6.2 shows the initial CSTs for the word aligned parallel corpus given in Table 6.1.

For calculating T, we subtract the chunk-start-index from each original word alignment. Chunk-start-index is equal to, first word position of the chunk in original sentence, minus one. For example, from Table 6.1 we get:

T=[around,230,million,people]=[7,8,9,10]

The first word of this chunk is “around”, which was in position 7. Subtracting 1, we get the T5 chunk-start-index is 6. Then we subtract this 6 from each word alignment, then we get final alignment, *T=[1,2,3,4]*.

However, not all English word is aligned with the Bengali words. For example CST3 has only one Bengali word, which is aligned with second word of the English chunk. Here “native language” is a phrase translated to “মাতৃভাষা” in Bengali. Our assumption is that Chunker will detect the phrases and keep it together in one chunk. Based on this assumption and the word alignment information, CSTs will be effective. If the chunker fails to identify phrases, it will be out of the assumption. In those exceptions, initial CSTs can generate wrong translation.

CST#	**English Chunk (C)**	**Bengali (S)**	**T**	**Align**	**Chunk-Start-Index**
CST1	[NP Bangla/NNP]	বাংলা	1	1	0
CST2	[VP is/VBZ]	হচ্ছে	1	2	1
CST3	[NP the/DT native/JJ language/NN]	মাতৃভাষা	2	4	2
CST4	[PP of/IN]	–এর	1	6	5
CST5	[NP around/RB 230/CD million/CD people/NNS]	প্রায় ২৩০ মিলিয়ন মানুষ	1 2 3 4	7 8 9 10	6
CST6	[ADVP worlwide/RB]	বিশ্বব্যাপী	1	11	10

Table 5.2: Example of initial CSTs

In Table6.2 CST# is the CSTs number for reference, C is the individual English Chunks, B is the corresponding Bengali Words, Align is same as Table 6.1, T contains the English- Bengali alignment information for each Bengali word. For each CSTs, we can get the original Align by adding Chunk-Start-Index and T.

6.1.3 CSTs Reorder

CSTs alignment stores the English word order and bengali word original sentence alignment information. So that from the initial CSTs the system can reorder the CSTs in Bengali word order.

In this step we generate the global alignment information from Initial CSTs as given in Table 6.2, based on the original word alignment as given in Table 6.1. For example, Table 6.3 shows the chunk alignment information produced from Table 6.1 and Table 6.2.

CT#	**CSTs**	**Global Alignment**
CCST1	CST1 CST2 CST3 CST4 CST5 CST6	CST6 CST1 CST2 CST5 CST4 CST3

Table 6.3: Example of CSTs global alignment

Figure 6.3 visualize the CSTs global alignment from Table 6.3.

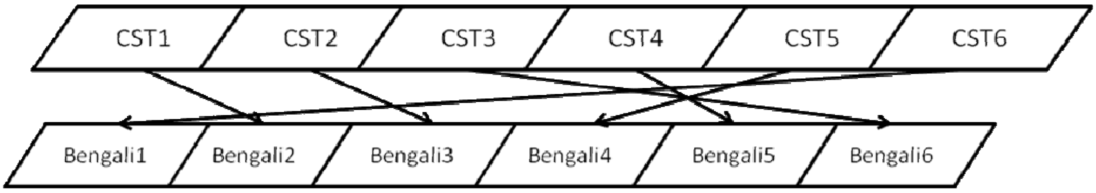


Figure 6.3: CSTs global alignment

6.1.4 CSTs Combination

In this step the system generates all possible chunk combinations. The goal is to match source language chunks with as many as possible CSTs. Without CSTs combinations, the system coverage will be low.

From CSTs alignment, as given in Table 6.3, system generates CSTs Combinations. It combines all sequential CSTs. For example in Figure 6.4, circles identified the sequential CSTs combination in Bengali word order. Here CST1 and CST2 can be combined as CCST2, because they are sequential in target language word order.

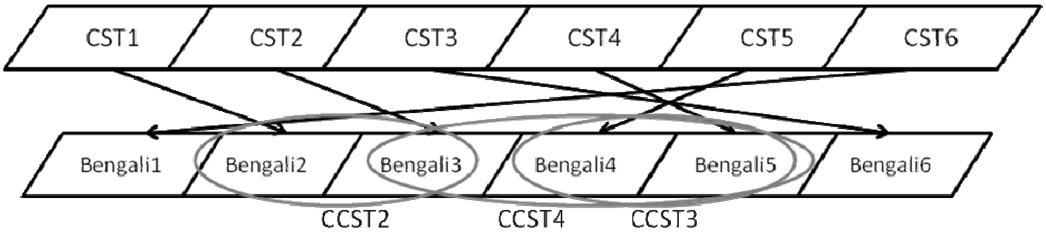


Figure 6.4: Chunk Alignment

Table 6.4 contains the Combined-CSTs (CCSTs) as shown in Figure 6.4. The system also produces CSTs combination in source language correspond in target language.

CT#	CSTs	Local Alignment
CCST1	CST1 CST2 CST3 CST4 CST5 CST6	CST6 CST1 CST2 CST5 CST4 CST3
CCST2	CST1 CST2	CST1 CST2
CCST3	CST4 CST5	CST5 CST4
CCST4	CST3 CST4 CST5	CST5 CST4 CST3

Table 6.4: Combined-CSTs examples

6.1.5 Generalize CST Using WordNet

In this step CSTs are generalized by using WordNet to increase the EBMT coverage. To generalize we only consider nouns, proper nouns and cardinal number (NN, NNP, CD in OpenNLP tagset). For each proper nouns we search in WordNet. If available we replace that NNP with <lexical filename> returned from the WordNet. For example WordNet return <noun.communication> for "Bangla". For cardinal number we simply CDs together to <noun.quantity>. We show example generalized CSTs produced using WordNet in Table 6.5.

CST#	English Chunk (C)	Generalized Chunk
CST1	[NP Bangla /NNP]	[NP <noun.communication>/NNP
CST5	[NP around/RB 230/CD million/CD people/NNS]	[NP around/RB <noun.quantity> people/NNS]

Table 6.5 : Generalized CSTs

Finally we get the CSTs database which has three tables: initial CSTs, generalized CSTs and CCSTs. From the example word-aligned parallel sentence of Table 6.1, system generated 6 initial CSTs, 2 Generalized CSTs and 4 Combined-CSTs.

6.2 Matching Algorithm for CSTs

Matching algorithm for CSTs has three components: search in CSTs, search in CCSTs and selecting CCSTs candidates. The Figure 6.5 shows the process of our proposed matching algorithm. The input is the English chunks from the source language sentence. At first the system find candidate CSTs for each SL chunks from initial CSTs. Search for each chunks using initial CST. Until all chunks are matched the system generalizes the input chunks and search in generalized CSTs. Finally the system selects best CSTs combination from all the CCSTs candidates.

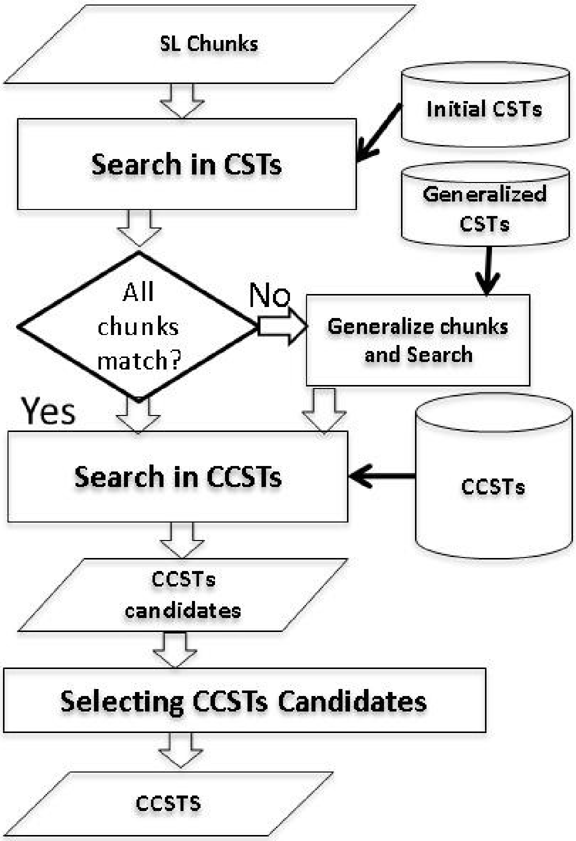


Figure 6.5: Matching Algorithm for CSTs

6.2.1 Search in CSTs

To search in CSTs our system first tries to find each chunk in initial CSTs. If it does not have exact match, it tries to find the linguistically related matches in generalized CSTs. Linguistically relations are determined by POS tags given in source-language chunks and the information provided by WordNet. Finally this step provides a set of matched CSTs. All SL chunks can be matched with at CSTs, generalized CSTs; or marked as OOV otherwise.

For example, we have 3 input chunks: [NP English/NNP][VP is/VBZ][NP the/DT native/JJ language/NN]. Second and third chunks are matched with CST2 and CST3 of initial CSTs in Table 6.2. But the first chunk [NP English/NNP], has no match. Then using WordNet the system generalized the input chunk “[NP English/NNP]” into “[NP <noun.communication>/NNP]”. It matched with CST1 of Table 6.5. This step returns a set of matched CSTs [CST1, CST2, CST3] and match level (as described in section 6.2.3).

6.2.2 Search in CCSTs

The second step is to search the matched CSTs in CCSTs. The system performs all order CSTs combina-tion search. And it returns CCSTs candidates. For the above example, it returns [CCST1, CCST2, CCST4,CCST5] because these CCSTs include at least one matched CST in

[CST1,CST2,CST3]. As this example if more than one CCSTs matches the CSTs, it returns all the CCSTs candidates, to select the best one in the next step.

6.2.3 Selecting CCSTs candidates

Finally in this step using our selection criteria we choose the suitable CCSTs. From the set of all CCSTs candidates this algorithm selects the most suitable one, according to the following criteria:

1. The more CSTs matched, the better;

2. Linguistically match give priority by following these ranks, higher level is better:

- Level 4: Exact match.
- Level 3: <lexical filename> of WordNet and POS tags match
- Level 2: <lexical filename> of WordNet match
- Level 1: Only POS tags match
- Level 0: No match found, all unknown words.

CHAPTER 7

UNKNOWN WORD TRANSLATION

As in our assumption, the main users of this EBMT will be monolingual people; they cannot read or understand English words written in English alphabet. However, with related word translation using WordNet and Transliteration can give them some clues to understand the sentence meaning. As Bangla language accepts foreign words, transliterating an English word into Bangla alphabet, makes that a Bangla foreign word. For example, in Bangla there exist many foreign words, so that user can identify those as foreign words.

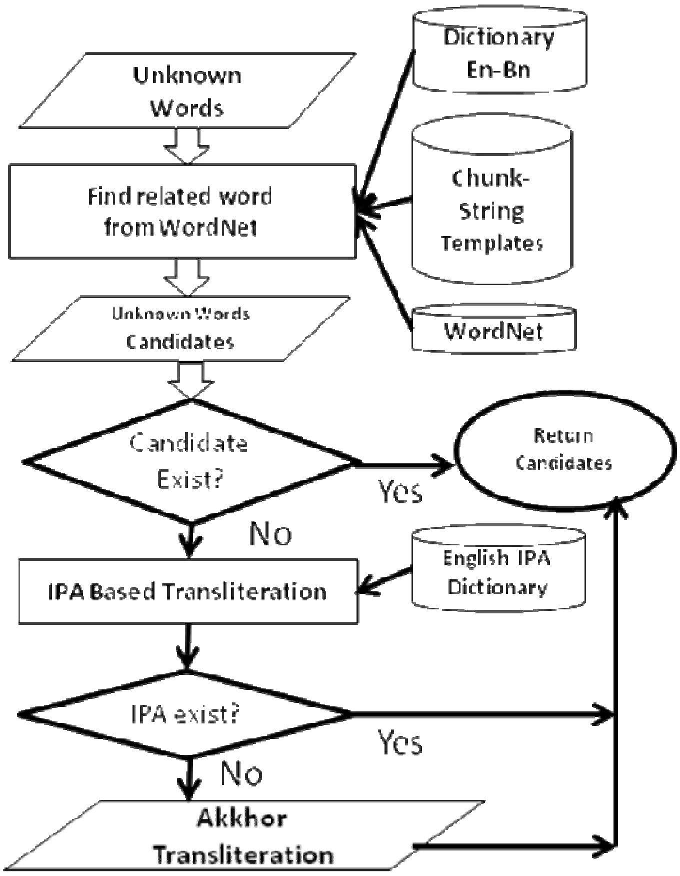


Figure 7.1: Steps of Unknown Word Translation

Figure 7.1 shows the unknown word translation process in a flow chart. Proposed system first tries to find semantically related English words from WordNet for the unknown words. From these related words, we rank the translation candidates using WSD technique and English-Bangla dictionary. If no Bangla translation exists, the system uses IPA-based-transliteration. For proper nouns, the system uses transliteration mechanism provided by Akkhor Bangla Software.

7.1 Find Sublexical Translations

For sublexical matching our system divide the unknown word into sublexical units and then find possible translation candidates from these sublexical units. For this the system use following steps:

(1) Find the possible sublexical units of the unknown word. For example, the unknown word “bluebird” gets divided into “blue” and “bird”.

(2) Extract sublexical translations and restrain translation choices.

(3) Remove less probable sublexical translations

(4) Output translation candidates with POS tags for the sublexical units of the unknown word.

From the set of all CSTs we select the most suitable one, according to the following criteria:

1. The more exact CSTs matched, the better;

2. Linguistically match give priority by following these ranks, higher level is better:

·Level 4: Exact match.

·Level 3: Sublexical unit match, <lexical filename> of WordNet and POS tags match

·Level 2: Sublexical unit match, <lexical filename> of WordNet match

·Level 1: Only POS tags match.

·Level 0: No match found, all unknown words.

7.2 Find Candidates from WordNet

Due to small English-Bangla parallel corpus availability, there is high probability for the MT system to handle unknown words. Therefore, it is important to have a good method for translating unknownwords. When the word has no match in the CSTs, it tries to translate using English WordNet and bilingual dictionary for English-Bangla. Input of this step is unknown words. For example “canine” is a unknown word in our system. Output of this process is the related unknown words translation.

7.2.1 Find Candidates from WordNet Synonyms

The system first finds the synonyms for the unknown word from the WordNet synsets. Each synset member becomes the candidate for the unknown word. WordNet provide related word for nouns, proper nouns, verbs, adjectives and adverbs. Synonymy is WordNet’s basic relation, because

WordNet uses sets of synonyms (synsets) to represent word senses. Synonymy is a symmetric relation between word forms. We can also use Entailment relations between verbs available in WordNet to find unknown word candidate synonyms.

7.2.2 Find Candidates Using Antonyms

WordNet provide related word for nouns, proper Antonymy (opposing-name) is also a symmetric semantic relation between word forms, especially important in organizing the meanings of adjectives and adverbs. For some unknown we can get the antonyms from WordNet. If the antonym exists in the dictionary we can use the negation of that word to translate the unknown word. For example, "unfriendly" can be translated as "not friendly". In Bengali to negate such a word we can simply add "না" (na) at the end of the word. So, "unfriendly" can be translated as "বন্ধুত্বপূর্ণ না" (bondhuttopurno na). It helps to translate unknown words like "unfriendly", which improves the machine translation quality.

Hyponymy (sub-name) and its inverse, hypernymy (super-name), are transitive relations between synsets. Because there is usually only one hypernym, this semantic relation organizes the meanings of nouns into a hierarchical structure. We need to process the hypernyms to translate the unknown word.

7.2.3 Find Candidates Using Hypernyms

For nouns and verbs WordNet provide hypernyms, which is defined as follows:

Y is a hypernym of X if every X is a (kind of) Y.

For example "canine" is a hypernym of noun "carnivore", because every dog is a member of the larger category of canines. Verb example, "to perceive" is an hypernym of "to listen". However, WordNet only provides hypernym(s) of a synset, not the hypernym tree itself. As hypernyms can express the meaning, we can translate the hypernym of the unknown word. To do that, until any hypernym's Bangla translation found in the English-Bangla dictionary, we keep discovering upper level of hypernym's. Because, nouns and verbs are organized into hierarchies, defined by hypernyms or is-a-relationships in WordNet. So, we considered lower level synset words are generally more suitable then the higher level synset words.

This process discovers the hypernym tree from WordNet in step by step. For example, from the hypernym tree of dog, we only had the "animal" entry in our English-Bengali dictionary. Our system discovered the hypernym tree of "dog" from WordNet until "animal".

Following is the discovered hypernym tree:

```
dog, domestic dog, Canis familiaris
=> canine, canid
     => carnivore
         => placental, placental mammal, eutherian mammal
                => mammal
                       => vertebrate, craniate
                           => chordate
                                  => animal    => ...
```

This process search in English-Bangla dictionary, for each of the entry of this hypernym tree. So at first we used the IPA representation of the English word from our dictionary, then using transliterating that into Bengali. Then system produce “a kind of X” - এক ধরনের X [ek dhoroner X]. For the example of “canine” we only had the Bengali dictionary entry for “animal” from the whole hypernym tree. We translated “canine” as the translation of “canine, a kind of animal”, in Bangla which is “ক্যানাইন, এক ধরনের পশু” [kjanain, ek dhoroner poshu].

Similarly, for adjectives we try to find “similar to” words from WordNet. And for Adverbs we try to find “root adjectives”.

Finally, this step returns unknown words candidates from WordNet which exist in English-Bangla dictionary.

Using the same technique described above, we can use Troponyms and Meronyms to translate unknown words. Troponymy (manner-name) is for verbs what hyponymy is for nouns, although the resulting hierarchies are much shallower. Meronymy (part-name) and its inverse, holonymy (whole-name), are complex semantic relations. WordNet distinguishes component parts, substantive parts, and member parts.

7.3 Rank Candidates

To choose among the candidates for the unknown word, we need to rank all the candidates. Especially polysemous unknown words need to select the adequate WordNet synset to choose the right candidate. The system perform Google search with the input sentence as a query, by replacing the unknown word with each candidate words. We add quotation marks in the input sentence to perform phrase searches in Google, to find the number of in documents the sentence appear together. If the input sentence with quotation mark returns less than 10 results, we perform Google search with four and two neighbor chunks. Finally, the system ranks the candidate words using the Google search hits information.

For example, the input sentence in SL is: This dog is really cool. The system first adds double quotation with the input sentence: “This dog is really cool”, which returns 37,300 results in Google. Then the system replaces the unknown word “dog” from discovered hypernym tree. Only for "This animal is really cool.", returned 1,560 results by Google. That is why “animal” is the second most suitable candidate for “dog”.However, other options "This domestic dog is really cool." or "This canine is really cool." etc. returns no results or less than 10 results in Google. So in this case we search with neighbour chunks only. For example, in Google we search with:

"This mammal is" returns 527,000 results;
"This canid is" returns 503,000 results;
"This canine is" returns 110,000 results;
"This carnivore is" returns 58,600 results;
"This vertebrate is" returns 2,460 results;
"This placental is" returns 46 results;
"This craniate is" returns 27 results;
"This chordate is" returns 27 results;
"This placental mammal is" returns 6 result;

Finally the system returns the unknown word candidates: mammal, canid, canine, carnivore, vertebrate, placental, craniates, chordate, placental mammal.

7.4 Final Candidate Generation

In this step, we choose one translation candidate. If any of the synonyms or candidate word exist in English-Bangla dictionary, the system translates the unknown word with that synonym meaning. If multiple synonyms exist then the entry with highest Google search hits get selected. English-Bangla dictionary also contains multiple entries in target language. For WSD analysis in target language, we perform Google search with the produced translation by the system. The system chooses the entry with highest Google hits as final translation of the unknown word. For example, for unknown word “dog”, animal get selected in our system. However, if there were no candidates, we use IPA-Based-Transliteration.

7.4.1 Transliterate if no candidate found from WordNet

When unknown word is not even found in WordNet, we use IPA-Based transliteration using the English IPA Dictionary as described in section VI.

However, when unknown word is not even found in the English IPA dictionary, we use transliteration mechanism of Akkhor Bangla Software. For example, for the word “Muhammod” which is a popular Bangla name, Akkhor transliterated into “মুহাম্মদ” in Bangla.

7.4.2 IPA-Based Transliteration

English words pronunciations in IPA obtained from the English IPA dictionary. Output for this step is the Bangla word transliterated from the IPA of the English word. In this step, we use following English-Bangla Transliteration map to transliterate the IPA into Bangla alphabet. Table 7.1, 7.2 and 7.3 shows our proposed English-Bangla IPA chart for vowels, diphthongs and consonants.

TABLE 7.1 : ENGLISH-BANGLA IPA CHART FOR VOWELS

Mouth narrower vertically	[i:] ই / ◌ি sleep /sliːp/	[I] ই / ◌ি slip /slIp/	[ʊ] উ / ◌ু book /bʊk/	[u:]উ / ◌ু boot /buːt/
	[e] এ / ে◌ ten /ten/	[ə] আ / ◌া after /aːftə/	[ɜ:] আ / ◌া bird /bɜːd/	[ɔ:] র্ bored /bɔːd/
Mouth wider vertically	[æ]এ্যা/ ◌্যা cat /kæt/	[^] আ /◌া cup / k^p/	[ɑ:] আ / ◌া car / cɑːr/	[ɒ] অ hot /hɒt/

TABLE 7.2: ENGLISH-BANGLA IPA CHART FOR DIPHTHONGS

[Iə] ইয়া/◌িয়া beer /bIər/	[eI] এই/ ে◌ই say /seI/	
[ʊə] উয়া/ ◌ুয়া fewer /fjʊər/	[ɔI] অয়/য়	[ə ʊ] ও / ে◌া no /nəʊ/
eə ঈয়া/ ◌ীয়া bear /beər/	[aI] ◌াই / আই high /haI/	[aʊ]আউ /◌াউ cow /kaʊ/

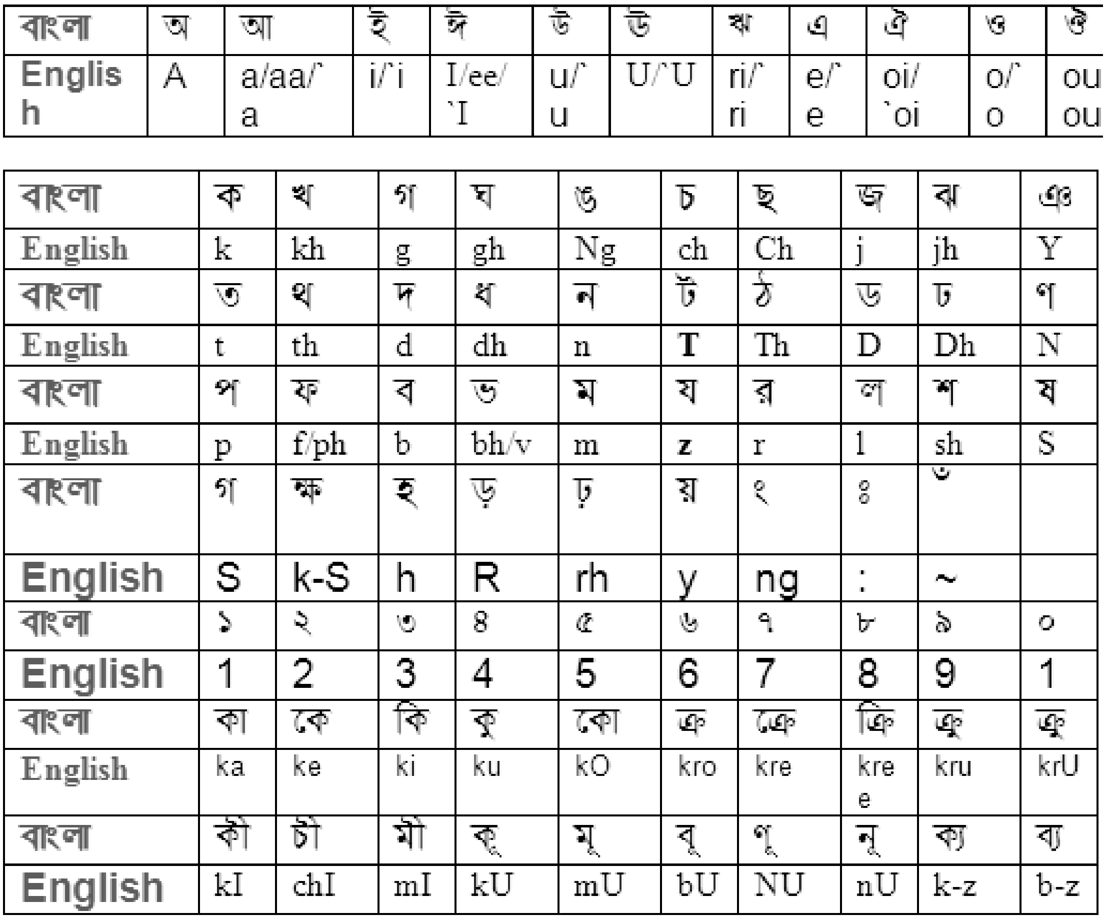
TABLE 7.3 ENGLISH-BANGLA IPA CHART FOR CONSONANTS

[p] প pan /pæn/	[b] ব ban /bæn/	[t] ট tan /tæn/	[d] ড day /deI/	[ʧ] চ chat /ʧæt/	[ʤ] জ judge /ʤ^ʤ/	[k] ক key /kiː/	[g] গ get /get/
[f] ফ fan /fæn/	[v] ভ van / væn/	[θ] থ thin /θIn/	[ð] দ than /ðæn/	[s] স sip /sIp/	[z] জ zip / zIp/	[ʃ] শ ship /ʃIp/	[ʒ] স vision /vIʒ^n/
[m] ম might /maIt/	[n] ন night /naIt/	[ŋ]◌ং/ঙ thing /θIŋ/	[h] হ height /haIt/	[l] ল light /laIt/	[r] র right /raIt/	[w] য় white /hwaIt/	[j]ইয়ে/◌িয়ে yes /jes/

Figure 7.2: English-Bengali IPA mapping

When unknown word is not even found in WordNet, we use IPA-Based transliteration using the English IPA Dictionary (Salam et. al., 2011). Output for this step is the Bangla word transliterated from the IPA of the English word. In this step, we use English-Bangla Transliteration map to transliterate the IPA into Bangla alphabet. From English IPA dictionary the system can obtain the English words pronunciations in IPA format. Output for this step is the Bengali word transliterated from the IPA of the English word. In this step, we use following English-Bengali Transliteration map to transliterate the IPA into Bengali alphabet. Figure 7.2 shows our proposed English-Bengali IPA chart for vowels, diphthongs and consonants. Using rule-base we transliterate the English IPA into Bangla alphabets. The above IPA charts leaves out many IPA as we are considering about translating from English only. To translate from other language such as Japanese to Banglawe need to create Japanese specific IPA transliteration chart. Using the above English-Bangla IPA chart we produced transliteration from the English IPA dictionary. For examples: pan(pæn): প্যান; ban(bæn): ব্যান; might(maIt): মাইট.

However, when unknown word is not even found in the English IPA dictionary, we use transliteration mechanism of Akkhor Bangla Software as given in Figure 7.3. For example, for the word “Muhammod” which is a popular Bangla name, Akkhor transliterated into “মুহাম্মদ”.



বাংলা	অ	আ	ই	ঈ	উ	ঊ	ঋ	এ	ঐ	ও	ঔ
English	A	a/aa/`a	i/`i	I/ee/`I	u/`u	U/`U	ri/`ri	e/`e	oi/`oi	o/`o	ou ou

বাংলা	ক	খ	গ	ঘ	ঙ	চ	ছ	জ	ঝ	ঞ
English	k	kh	g	gh	Ng	ch	Ch	j	jh	Y
বাংলা	ত	থ	দ	ধ	ন	ট	ঠ	ড	ঢ	ণ
English	t	th	d	dh	n	**T**	Th	D	Dh	N
বাংলা	প	ফ	ব	ভ	ম	য	র	ল	শ	ষ
English	p	f/ph	b	bh/v	m	**z**	r	l	sh	S
বাংলা	গ	ক্ষ	হ	ড়	ঢ়	য়	ং	ঃ	ঁ	
English	S	k-S	h	R	rh	y	ng	:	~	
বাংলা	১	২	৩	৪	৫	৬	৭	৮	৯	০
English	1	2	3	4	5	6	7	8	9	1
বাংলা	কা	কে	কি	কু	কো	ক্র	ক্রে	ক্রি	ক্রু	ক্রূ
English	ka	ke	ki	ku	kO	kro	kre	kree	kru	krU
বাংলা	কী	চী	মী	কূ	মূ	বূ	ণূ	নূ	ক্য	ব্য
English	kI	chI	mI	kU	mU	bU	NU	nU	k-z	b-z

Figure 7.3– Akkhor phonetic mapping for Bengali alphabets

7.5 Translation Generation

In this EBMT architecture we used Rule-Based generation method. Using dictionary and WordNet rules, we can accurately translate the determiners in Bengali. For translating determiner we adapted (Salam et al. 2009) proposals to use WordNet.

To reorder the CSTs for partial match in CCSTs, we remove the unmatched CSTs. Based on the morphological rules we change the expression of the words.

Here WordNet provided required information to translate polysemous determiners accurately. The system compared with the <lexical filename > of WordNet for the word NN. If the word NN is "<noun.person>", then determiner "a" will be translated as "ekjon". Otherwise "a" will be translated as "ekti".

For example "a boy" should be translated to "ekti chele" as boy is a person. "ekkhana chele" is a wrong translation, because "ekkhana" can be used only for NNs which is not a person.

For Bengali word formation we have created morphological generation rules especially for verbs. These rules are constructed by human.

Chapter 8

Experiment

We did wide-coverage and quality evaluations for the proposed EBMT with CSTs, by comparing with baseline EBMT system. Wide-coverage evaluation measures the increase of translation coverage. Quality evaluation measures the translation quality through human evaluation.

Baseline system architecture has the same components as described in Figure 1, except for the components inside dotted rectangles. Matching algorithm of baseline system is that not only match with exact translation examples, but it can also match with POS tags. The Baseline EBMT use the same training data: English-Bengali parallel corpus and dictionary, but does not use CSTs, WordNet and unknown words translation solutions.

Grammatical Structures of test-set sentences

English sentences in our test-set can be classified in four types: Declarative, Imperative, Interrogative and Exclamatory sentences. These sentences can also be classified using following complexity types: Simple, Compound, Complex and Compound-Complex. Current EBMT system performance depend on the quality of English chunker.

Currently from the training data set of 2,000 word aligned English-Bengali parallel corpus, system generated 15,356 initial CSTs, 543 Generalized CSTs and 12,458 Combined-CSTs.

The development environment was in windows using C Sharp language. Out test-set contained 336 sentences, which are not same as training data. The test-set includes simple and complex sentences, representing various grammatical phenomena. We have around 20,000 English-Bengali dictionary entries.

8.1 Wide-Coverage Evaluation

We calculated the rate of generalized CSTs usage to evaluate the achievement of wide-coverage. To match the English input chunks, baseline EBMT use translation examples and POS matching mechanism from the training data. On the other hand, proposed EBMT use CSTs to match the English input chunks.

Table 6 shows the contribution of CSTs to achieve wide-coverage. Here wide-coverage = No. of Matched English chunks / No. of all English chunks in test-set. CSTs improved the wide-coverage by 57 points.

System Modules	wide-coverage
Baseline EBMT	23%
Proposed EBMT with CSTs	80%

Table 6: Wide-Coverage Comparison

8.2 Quality Evaluation

8.2.1 CSTs Evaluation

Quality evaluation measures the translation quality through human evaluation. Table 7 shows the human evaluation of the proposed EBMT system with CSTs only.

Translation Quality	Grade	EBMT+ CSTs	Google
Perfect Translation	A	25.60	19.00
Good Translation	B	38.69	30.00
Medium Translation	C	19.64	27.00
Poor Translation	D	16.07	24.00
Total		**100%**	**100%**

Table7: Human Evaluation using same testset

Translation Quality	Word Selection	Word Order	Functional Word Usage
Perfect Translation	YES	YES	YES
Good Translation	YES	YES	NO
Medium Translation	YES	NO	YES/NO
Poor Translation	NO	NO	NO

Table8: Human Evaluation quality explanation

Table 8 shows the explanation of translation quality used in our human evaluation process. Word selection means whether the system could choose a correct word candidate. Word order measures whether the words position in the translated sentence is grammatically correct. Functional word usage means whether the system could choose a correct functional word. Considering these quality elements we have evaluated the translation quality.

Perfect Translation means there is no problem in the target sentence. Good Translation means the target sentence is not grammatically correct because of wrong functional word, but still understandable for human. Medium means there are several mistakes in the target sentence, like wrong functional word and wrong word order. So human cannot understand the translated sentences in medium category. Poor Translation means there are major problems in the target sentence, like non-translated words, wrong word choice and wrong word order.

Only perfect and good translations were "acceptable". Because even though the system choose the correct word without generating the correct word order the translated sentence will be grammatically incorrect and may not be understandable.

Currently 64.29% of the test-set translations produced by the system were acceptable, produced by the system with proposed CSTs only. Around 48.81 points of poor translation produced by EBMT Baseline was improved using the proposed system with CSTs.

The identified main reasons for improving the translation quality is our solution using CSTs generalization and sub-sentential match. Because of these contributions of CSTs some test-set sentence improved from "poor" or "medium" translation to "acceptable" translation.

We observed some drawbacks of using CSTs with generalization using WordNet as well. Sometimes our system chooses the wrong synset from the WordNet. As a result, some test-set still produced "poor" translation.

8.2.2 Unknown Words Evaluation

We also did quality evaluation for our unknown words solution. Table 9 shows the human evaluation of the EBMT system with CSTs and unknown word solution. Currently 67.85% of the test-set translations were acceptable, produced by the system with proposed CSTs and unknown words solutions. Comparing with EBMT+CSTs, unknown words mechanism improved translation quality by 3.56 points in human evaluation. We also compare our system with Google translate which is the most popular MT system for English-Bengali language pair.

Translation Quality	Grade	EBMT+CSTs+Unknown Words
Perfect Translation	A	30.95
Good Translation	B	36.90
Medium Translation	C	18.75
Poor Translation	D	13.39
Total		100.00

Table 9: Human Evaluation of Unknown words using same test-set

Our EBMT could translate better than Google because of our novel unknown words translation mechanism. As we used same test-set, the result of Google MT is same for both Table 7 & 9.

Table 10 shows sample translation examples produced by Google and EBMT with CSTs, unknown words solution. It also shows the translation quality in bracket (A,B,C,D: Perfect, Good, Medium, Poor).

#	English	EBMT+CSTs+Unknown Words	Google
1.	Are you looking for an aardvark?	আপনি কি আর্ডভার্ক, এক ধরনের পশু খুঁজছেন?(A)	আপনি যদি একটি Aardvark খুঁজছেন? (D)
2.	This dog is really cool.	ডগ, এক ধলনেল পশু আসলেই দারুন (A)	এই কুকুর সত্যিই শান্ত. (C)
3.	WordNet is a..	শব্দজাল হচ্ছে.. (A)	একটি ওয়ার্ডনেট হয় .. (B)
4.	Sublexical units of a word	শব্দের উপ-আভিধানিক অংশ (A)	একটি শব্দের Sublexical ইউনিট (D)
5.	This is a bluebird	এটা নীলপাখি .. (A)	এটি একটি Bluebird হয় (D)

Table10: Human Evaluation of unknown words using same testset

As "aardvark", "Sublexical" and "bluebird" are unknown words, Google MT produced poor translation for #1, #4 and #5. However our proposed solution could generate "good" quality translations. All these examples demonstrate the effectiveness of our proposed solution for translating unknown words.

8.3 Wide-Coverage of Adequate Determiner Evaluation

As we used WordNet to translate using adequate determiner, we measured the increase of translation coverage as following.

$$\text{wide} - \text{coverage} = \frac{\text{No. of system generate adequate determiner}}{\text{No. of all adequate determiner}}$$

(from example Human evaluation sentences)

System Modules	wide-coverage
Baseline EBMT	24%
Proposed EBMT with WordNet	65%

Table 11: Wide-Coverage Comparison

Table 11 shows the EBMT system performance improvement for the test data of 336 sentences. In these test sentences we had 107 adequate determiners. The baseline EBMT produced 34 adequate determiners, which is 24% of all adequate determiners. The proposed EBMT produced 93 adequate determiners, which is 65% of all adequate determiners. Our proposed EBMT system improved the wide-coverage of adequate determiners by 41 points. We found generalized CSTs are also effective for achieving wide-coverage in translating determiners.

Chapter 9

Expansion using UNL Ontology

To develop a common language, it is essential to have enough vocabulary to express all the concepts contained in all the world languages. Those vocabularies can only be developed by native speakers and should be defined by formal ways. Considering the situation, at this moment Universal Networking Language (UNL) is the best solution as the common language, and Universal Words (UWs) are the most promising candidates to represent all the world concepts in different languages. However, UWs itself are formal and not always to be understandable for human. To ensure every language speakers can create the correct UWs dictionary entry, we need to provide the explanation of UWs in different natural languages for humans. As there are millions of UWs, it is very expensive to manually build the UWs explanation in all natural languages. To solve this problem, this research proposes the way to auto generate the explanation of each UWs using the semantic backgrounds provided by UNL Ontology. These explanations can be useful for translating unknown words in our proposed EBMT architecture.

9.1 UNL Ontology

'''UNL Ontology''' is a semantic network with hyper nodes of [[Universal Words]] of [[Universal Networking Language]] (UNL). It contains [[UW System]] which describes the hierarchy of the UWs in lattice structure, all possible semantic co-occurrence relations between each UWs and UWs definition in UNL. With the property inheritance based on UW System, possible relations between UWs can be deductively inferred from their upper UWs and this inference mechanism reduces the number of binary relation descriptions of the UNL Ontology. In the topmost level UWs are divided into four categories: adverbial concept, attributive concept, nominal concept and predicative concept. UNL Ontology provides the semantic background of word using word hierarchy, semantic co-occurence with deductive inference, and meaning definition in UNL graph (knowledge representation). UNL Ontology is available at [[http://www.undl.org/unlsys/uw/UNLOntology.html UNL Center]].

==Universal Words (UWs)==
UWs constitute the UNL vocabulary. They are labels for concepts, syntactic and semantic units to form UNL Expressions. A combination of a set of UWs - linked with each other through relations and modified by attributes - expresses the meaning of a sentence. A UW of UNL is defined in the following format:

<uw> =:: <headword>[<constraint list>]

A headword of a UW is an English expression, a word, a compound word, a phrase or a sentence of English. If the meaning of a headword is unique, the headword itself becomes a UW. Otherwise, constraints are attached to the headword to make more specific UWs. If a UW consists of a headword only, it is called a basic UW ･

For example, hear(icl>perceive(agt>person,obj>thing)) is a UW where the headword is "hear" and the constraint list is "(icl>perceive(agt>person,obj>thing))"

==Universal Words (UWs) Definition==
<uw> =:: <headword>[<constraint list>][UW Definition]

==UNL Knowledge Base (UNLKB)==
The UNLKB is a semantic network comprising every directed binary relation between UWs. All binary relations of the UNL KB are in the following format: 'relation(UW1, UW2)=c', where 'c' is the degree of certainty, which has the value 0 (impossible) or from 1 to 128 (certain). This binary relation means UW1 takes UW2 as the relation in certainty value c ･ or UW2 plays the role specified by the relation to UW1 in certainty value c ･ In the UNLKB, the semantics of UWs are defined using the UW system and linguistic knowledge of concepts is provided also based on the UW System.

'''The UNLKB Defines Semantics of UWs:''' A UW is a label for a concept. Concepts labeled by UWs are defined by describing the set of possible relations that each concept can have with other concepts in UNLKB. Definitions of possible relations of a concept with other concepts describe the behavior of the concept. This behavior is the property of a concept in the sense that the descriptions of behavior characterize the concept and provide enough information for understanding the semantic structure of a sentence which include the concept.

'''The UNLKB Provides Linguistic Knowledge of Concepts:''' The behavior of a concept is considered as linguistic knowledge on the concept. This knowledge is used to provide semantic structure of sentences of natural languages. For example, an author "is a person who can take various actions that a person can take, such as writing something and something might be a book, and so forth". This level of knowledge is necessary to provide the semantic background of natural language sentences. Further knowledge, for example real world knowledge, will be established based on this linguistic knowledge, using the UWs.

== UW System==

UW System contains the hierarchy relationships of UWs using icl ▪ (subclass of), iof ▪ (instance of), and iqu ▪ (equivalent to). It is possible to perform property inheritance and allows the substitutability of lower UWs with super-class UWs. Here, lower UWs can inherit the properties of upper UWs, and on the other hand, upper UWs can replace lower UWs for generlization.

The UW System allows having multiple superordinate (super class) concepts of UWs. The hierarchy of the UW System is a lattice network. A UW can inherit different sets of properties by linking it to different upper UWs that have different properties.

==Linguistic Knowledge on Concepts==

UWs are the label for concepts. In the UNL Ontology each UWs are defined by describing the set of possible relations it can have with other UWs. The set of possible relations of a concept determine the behavior of the concept. This behavior is the property of a concept in the sense that the set of possible relations characterize the concept and provide enough information for understanding the semantic structure of a sentence when in which the concept is included.

==Semantic Knowledge on Concepts==
Necessary and sufficient conditions for belonging to the set (class) defined by a concept are necessary knowledge for reasoning. This knowledge is considered semantic knowledge about concepts. In the UNL Ontology, every UW is given a concept definition. A concept definition is an intensional definition of the concept, consists of by a set of binary relations that specify all the essential properties of the concept. For instance, the definition of bachelor is "unmarried man" . and the definition of author： "is a person who writes books or a person who wrote a particular book"

Concept definitions are provided in the form of UNL Expressions. A UNL Expression of a concept definition is linked with correspondent UW as a hyper-node. The purpose of concept definitions is to provide knowledge of concepts in connection with other concepts that can specify the concepts. This knowledge is indispensable for reasoning in information retrieval

Figure 9.2 shows the topmost level of partial UNL Ontology where the black directed lines represent "icl" relation and dotted directed lines represent "agt" relations. In Figure 1 we only expanded partial "nominal concept" until "dog(icl>mammal)" to give a brief overview of the UNL Ontology. In UNL Ontology each UWs can have incoming and outgoing relations. For example in Figure 9.1 "animal(icl>living thing)" has two incoming relations, "agt" from "eat(agt>animal,obj>food)", and "icl" from "volitional thing". "animal(icl>living thing)" has only one outgoing relation "icl" to "mammal(icl>animal)". As possible relations between lower UWs are deductively inferred from their upper UWs, we can infer that "mammal(icl>animal)", "canine(icl>mammal)" and "dog (icl>mammal)" also has an incoming relation "agt" from "eat(agt>animal,obj>food)".

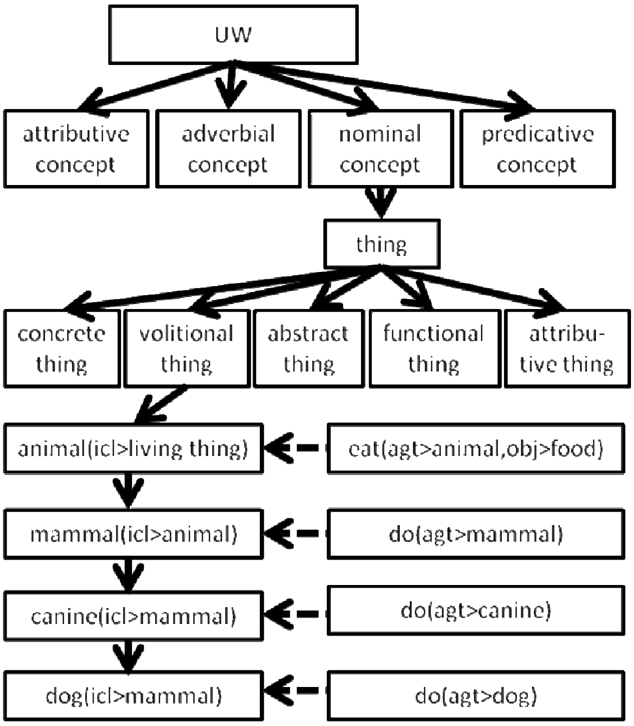


Figure 9.2. ***UW System (hierarchy) in UNL Ontology***

A. Previous editing process

UW dictionary editors have to provide the UWs relations and required dictionary entries. The current process flow has been described using Figure 9.2. When the editor find a new concept the first step is to make a master definition in UNL from the definitions in natural language. Such definitions can be taken from standard language dictionary. From this master definition the editor defines the UW. Then finally the UW has been registered by the dictionary editor in UNL Ontology.

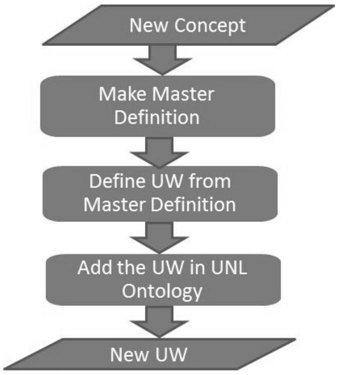


Figure 9.2. ***. Old work flow for making new UW***

B. New editing process

Figure 9.3 shows the new work flow for the dictionary editors to update the UWs definition. For this the editor choose a UW first. Then the editor can use this visualization system to evaluate the quality of UWs and their relations. Based on their understanding editors can change the UWs relations using the systems graphical interface.

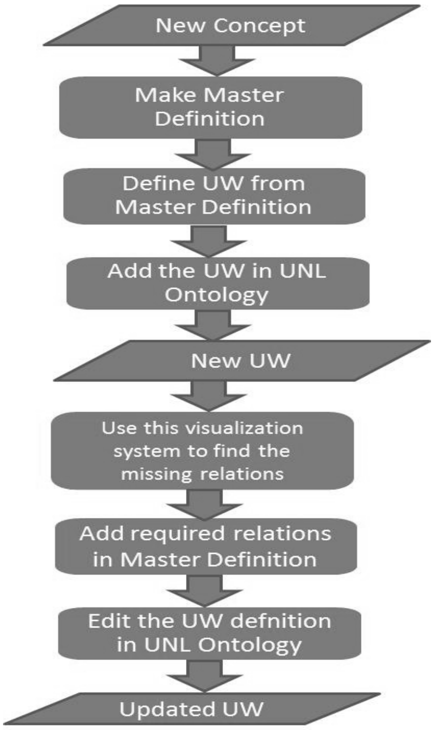

New Concept
Make Master Definition
Define UW from Master Definition
Add the UW in UNL Ontology
New UW
Use this visualization system to find the missing relations
Add required relations in Master Definition
Edit the UW defnition in UNL Ontology
Updated UW

Figure 9.3. ***New work flow for making new UW***

C. UNL Ontology Visualization

UNL Ontology contains millions of UWs and relations between them. That is why it is almost impossible to display the complete UNL Ontology using one single map. However, the potentially huge amount of nodes and arcs could be organized in a meaningful spatial layout, alongside artificially added spatial features: such as color codes or place marks. The resulting visualization will be a mix between a classic node-link diagram and a so-called infographic, a graphic way to present information that abstracts and represents many qualitative and quantitative aspects of a subject in a carefully studied design.

To retrieve semantic background information users need to navigate the relations of UWs. This is often difficult for the users to browse UNL ontology, possible to lost the track for finding UW relations in raw data format. To address these problems, we have designed an interactive web-based system for UNL ontology visualization. Instead of leaving the users to manually traverse the UNL ontology, the developed system lets user visualize the information as it requires. The interactive ontology visualization encodes a number of properties that help users to see the relations of UWs to get the semantic background. And users remain oriented while navigating the ontology through the web browser. It helps the users to provide visual representation of the ontology for better understanding. The system can be accessed from any popular web browser such as Internet Explorer, Mozilla Firefox etc. The system also requires the browser to support Javascript.

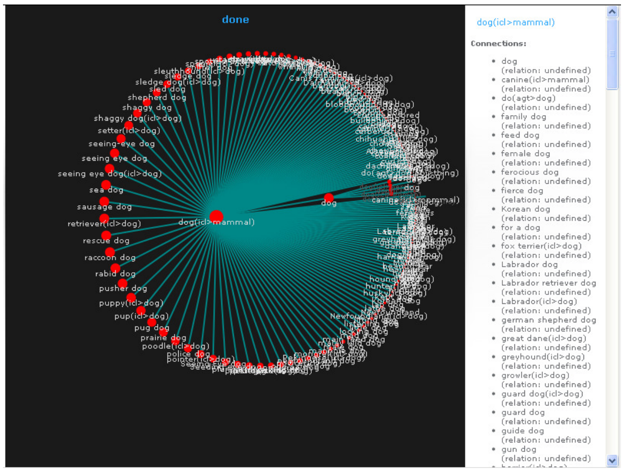
For example the user wants to know about the semantic relations for the UW "dog(icl>mammal)". As this tree includes all UW relations, it is difficult to understand from the raw data of UNL Ontology tree . But in this case user only needs to know the relations shown in Figure1, which only includes the relations related to the given UW. That is why the developed system retrieves all the deductively inferred relations and visualizes using a graph.

For this, we first discover Concept Map for the given UW. A Concept Map contains all direct and deductively inferred relations for one particular UW from the UNL Ontology. Archs of this graph are the relations of UNL Ontology. In UNL Ontology each relation is connected from "fromUW" to "toUW". Starting from a given UW we discover the Concept Map graph which includes deductively inferred relationships. A maximum search depth is established to limit the size of the graph.

To discover the Concept Map graph from UNL Ontology user has to give a particular UW. First the algorithm adds that UW into the Concept Map graph. For each outgoing relation from that UW, it add toUW into the Concept Map and then recursively call Concept Map(toUW) to discover the relations from toUW. Then for each incoming relationship it adds the fromUW with relation into the Concept Map graph. As UNL Ontology contains huge number of UWs and relationships, user can set the limit of the Concept Map graph to produce more meaningful and specific information. So the algorithm keeps discovering the graph until it reaches maximum search depth or if it reaches the topmost UW. Finally it returns the Concept Map graph which contains all the UW relations. Moreover, user can customize the limits to redraw the graph according to their own requirements.

9.1.1. Circle Visualization

Circle visualization shows the input UW in the center and the related UWs around by forming a circle. Figure9.4 shows a sample screenshot of the circle visualization for UNL ontology, which it shows all the relations for the UW of "dog(icl>mammal)". Each red node denotes a UW and each arc denotes a relation between the UWs.



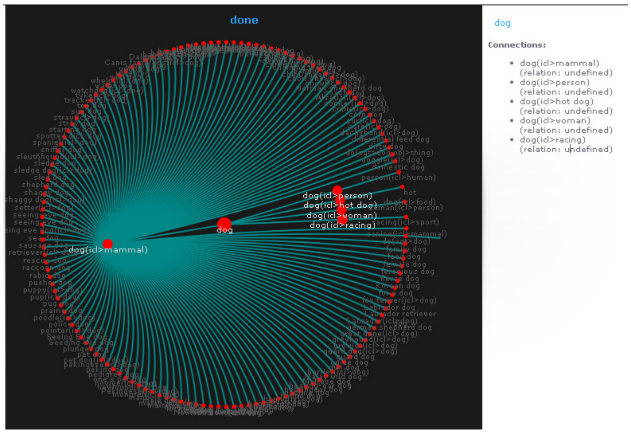
dog(icl>mammal)

Connections:

- dog
  (relation: undefined)
- canine(icl>mammal)
  (relation: undefined)
- do(agt>dog)
  (relation: undefined)
- family dog
  (relation: undefined)
- feed dog
  (relation: undefined)
- female dog
  (relation: undefined)
- ferocious dog
  (relation: undefined)
- fierce dog
  (relation: undefined)
- Korean dog
  (relation: undefined)
- for a dog
  (relation: undefined)
- fox terrier(icl>dog)
  (relation: undefined)
- Labrador dog
  (relation: undefined)
- Labrador retriever dog
  (relation: undefined)
- Labrador(icl>dog)
  (relation: undefined)
- german shepherd dog
  (relation: undefined)
- great dane(icl>dog)
  (relation: undefined)
- greyhound(icl>dog)
  (relation: undefined)
- growler(icl>dog)
  (relation: undefined)
- guard dog(icl>dog)
  (relation: undefined)
- guard dog
  (relation: undefined)
- guide dog
  (relation: undefined)
- gun dog
  (relation: undefined)

Figure 9.4. ***Screenshot of the web based UNL Ontology viewer***

Figure 9.5 shows the limitation of this approach for displaying more than 50 UWs. It is difficult for human to read the UWs in this visualization technique.



dog

Connections:

- dog(icl>mammal)
  (relation: undefined)
- dog(icl>person)
  (relation: undefined)
- dog(icl>hot dog)
  (relation: undefined)
- dog(icl>woman)
  (relation: undefined)
- dog(icl>racing)
  (relation: undefined)

Figure 9.5. ***Screenshot of the web based UNL Ontology viewer***

9.1.2. Tree Visualization

Figure 9.6 shows a sample screenshot of the Tree visualization of UNL ontology, which shows all the relations for the UW of "dog(icl>mammal)". Each box denotes a UW and each arc denotes a relation between the UWs.

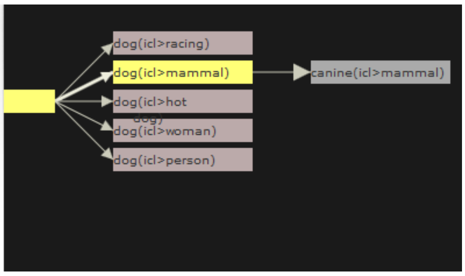


Figure 9.6. ***Screenshot of the Tree visualization for UNL Ontology***

9.1.3. Circle-Step Visualization

Figure 9.7 shows a sample screenshot of the Circle-Step visualization, which shows all the relations for the UW of “dog(icl>mammal)”. Each node denotes a UW and each arc denotes a relation between the UWs.

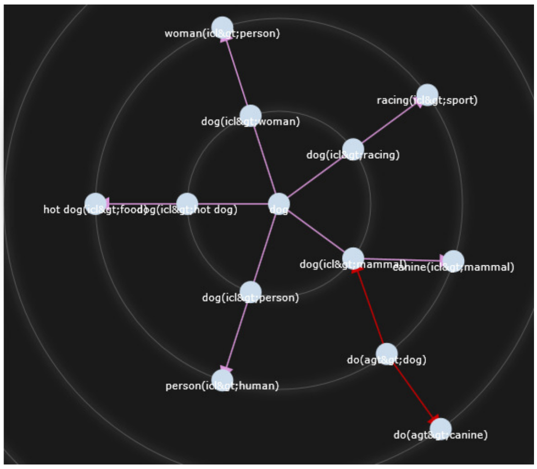


Figure 9.7. ***Screenshot of the Circle-Step visualization for UNL Ontology***

From raw data, it is difficult for human to visualize the UNL ontology. Here we described the web system to visualize UNL Ontology. It will help the users of UNL ontology by providing an interactive web interface to access the relations of UWs.

9.2 Universal Words Explanations

The system framework for automatic generation of the UWs explanation is illustrated in Figure 9.8. The input of this system is one UW and the output of the system is the explanation of that UW in a natural language such as English, Japanese etc. For the given UW, the system first discover a SemanticWordMap, which contains all direct and deductively inferred relations for one particular UW from the UNL Ontology. So input of this step is one UW and output of this step is the WordMap graph.

In next step we convert the WordMap graph into UNL using conversion rules. This conversion rules can generate "From UWs only" and "From UNL Ontology", based on user's requirement. So input of this step is the WordMap graph and Output is the UNL expression.

In the final step we describe in natural language by converting the UNL expression using UNL DeConverter, provided by UNL Explorer.

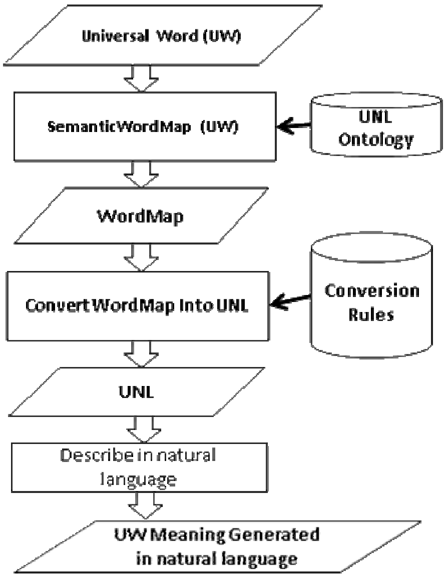


FIGURE 9.8 – UWs explanation generation steps

9.2.1 SemanticWordMap

To auto generate the UWs meaning, we need to get the deductively inferred relationship for a particular UW. We first discover a graph which we call SemanticWordMap. It contains all direct and deductively inferred relations for one particular UW from the UNL Ontology. Edges of this graph are the relations of UNL Ontology. In UNL Ontology each relation is connected from "fromUW" to "toUW". Starting from a given UW we discover the SemanticWordMap graph which includes deductively inferred relationships. A maximum search depth is established to limit the size of the graph.

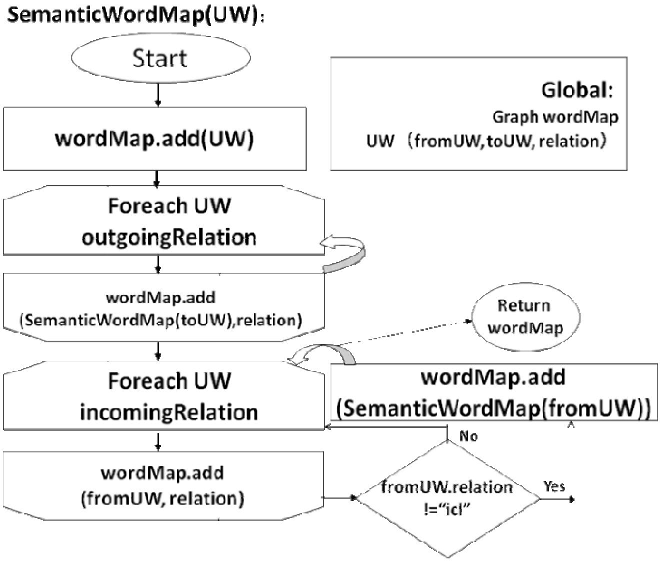

SemanticWordMap(UW):
Start
wordMap.add(UW)
Global:
Graph wordMap
UW (fromUW,toUW, relation)
Foreach UW outgoingRelation
wordMap.add (SemanticWordMap(toUW),relation)
Return wordMap
Foreach UW incomingRelation
wordMap.add (SemanticWordMap(fromUW))
No
wordMap.add (fromUW, relation)
fromUW.relation !="icl"
Yes

FIGURE 9.9: SemanticWordMap algorithm

To discover the SemanticWordMap graph from UNL Ontology user has to give a particular UW. First the algorithm adds that UW into the wordMap graph. For each outgoing relation from that UW, it add toUW into the wordMap and then recursively call SemanticWordMap(toUW) to discover the relations from toUW. Then for each incoming relationship it adds the fromUW with relation into the wordMap graph. If the relationship is not “icl”, it adds the expanded graph by recursively calling SemanticWordMap(fromUW). As UNL Ontology contains huge number of UWs and relationships, we have a heuristic approach to limit the SemanticWordMap graph to produce meaningful and specific information. So the algorithm keep discovering the graph until it reach maximum search depth or if it reach the topmost UW. Finally it returns the wordMap graph which contains all the UW relations.

For example, Figure 9.10 shows the partial SemanticWordMap for dog(icl>mammal). The output of this first step is the SemanticWordMap discovered from UNL Ontology. Here dotted arrows represent “agt” relations and black arrows are “icl” relations.

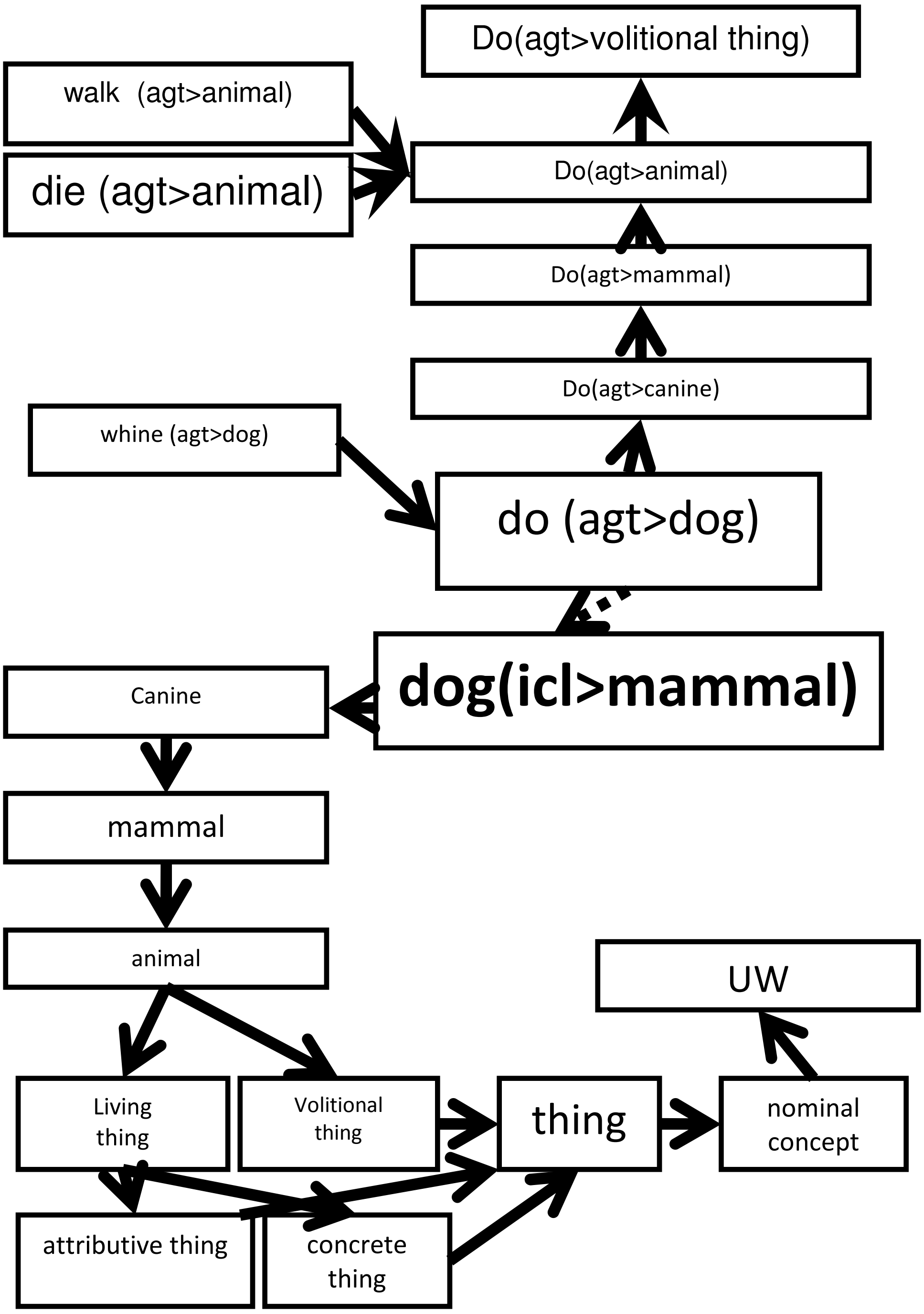


FIGURE 9.10: Partial SemanticWordMap for dog(icl>mammal)

9.2.2 Convert SemanticWordMap into UNL

In this step we convert SemanticWordMap relations into UNL using some general-ized rules. We first categorize the UWs into several categories such as “do”, “is-a”, “occur” and “be”. In general “do” cetegories represent actions, “is-a” represent fea-tures, “occur” represents changes and “be” represents status.

Due to the property inheritance characteristic of the UNL Ontology, possible rela-tions between lower UWs are deductively inferred from their upper UWs. Using SemanticWordMap we deductively infer the relationship with dog(icl>mammal). For example from Figure 3 we can say that UWs walk(agt>animal) and die(agt>animal) are related with dog(icl>mammal) as well. Table I shows the categorized relations from SemanticWordMap for the UW dog(icl>mammal), and the generated description categorized into several UW relationship types.

UW	Categorized from SemanticWordMap	
	UW Categories	*Description*
DOG (icl>mammal)	do	whine, walk, die….
	Is-a	canine, mammal, animal, ..

TABLE I. CATEGORIZED RELATIONS FOR DOG(ICL>MAMMAL)

For example, Figure 9.11 shows UWs relation derived from SemanticWordMap. To convert these category UWs relations into UNL, we use the following "Rule 1":

(Rule 1: do) If (isaKindof(UW2,"do"))

agt(UW3:08.@entry.@ability,UW1:00.@topic)

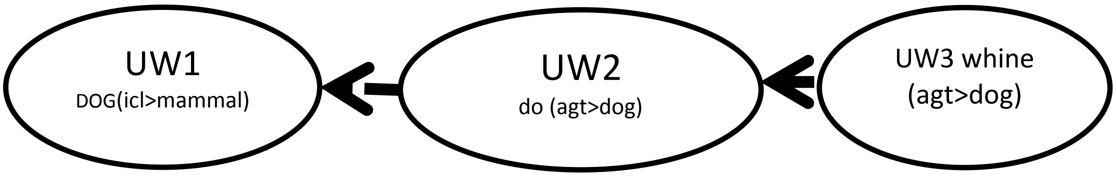


FIGURE 9.11: "do" relations derived from SemanticWordMap

Rule 1 check whether UW2 is related with "do" by using "*isaKindof* (UW2,"do"). For example if isaKindof("do (agt>dog)","do") = TRUE, then the generated UNL is:

agt(whine(agt>dog):08.@entry.@ability,dog(icl>mammal):00.@topic)

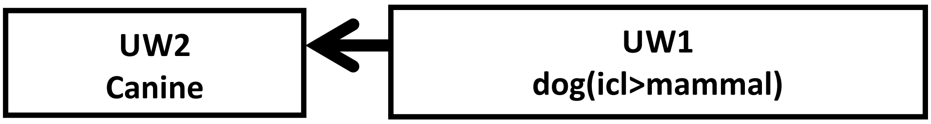


FIGURE 9.12: icl relations derived from SemanticWordMap

Figure 9.12 shows "icl" relation derived from SemanticWordMap. To convert this UWs into UNL we use following "Rule 2":*<u>(Rule 2: is-a)</u> If (isaKindof(UW1,UW2)) then icl(uw1:09, uw2:0F)*

Rule 2 check whether UW1 has "icl" relationship by using "*isaKindof* (UW1,UW2).

For example isaKindof("dog (icl>mammal)","canine*(icl>mammal)*") = TRUE, so the generated UNL is: *icl(dog(icl>mammal):09, canine(icl>mammal): 0F)*

The above mechanism works for UWs under “nominal concepts”. For other types of UWs such as “attributive concepts” we need to use different set of rules. For the UW write(agt>person,obj>report), we can get the partial SemanticWordMap as shown in Figure 9.12.

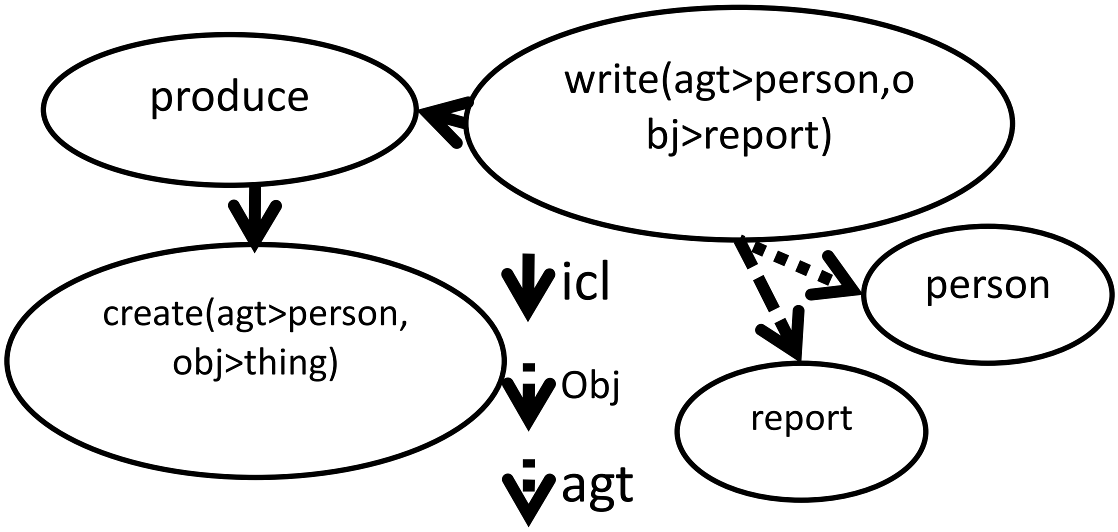


FIGURE 9.13: UNL Relations from UWs only

Figure 9.13 show that using only UWs, we can get UNL expression for “Person write a report”. However in the meaning of the UW we should not use that concept. Instead we can use immediate higher UW concept. So in this case instead of “write” we can use “produce”. By replacing “person” with the upper concept “someone”, we can get the UNL expression for “Someone produce a report”. In this way, using different rules the system can convert SemanticWordMap relations into UNL.

9.2.3 Describe in natural language

Finally, we used UNL DeConverter to convert our generated UNL expressions into natural languages. UNL DeConverter is a language independent generator that provides a framework for syntactic and morphological generation as well as co-occurrence-based word selection for natural collocation. It can deconvert UNL expressions into a variety of native languages, using a number of linguistic data such as Word Dictionary, Grammatical Rules and Co-occurrence Dictionary of each language. We used UNL DeConverter to convert the UNL into several natural languages. In this paper we only report the English results.

9.2.4 Implementation in UNL Explorer

For experiment we implemented the proposed method in UNL Explorer. Our implementation could successfully produce 1466598 UWs explanation in UNL. Table II shows some sample UWs explanation generated by the proposed method. Here, we only reported sample explanations in English and Japanese, together with the UNL expression.

Universal Word	Explanation Generated from UNL Ontology in Different Languages		
	English	*Japanese*	*UNL*

write(agt>person,obj>report)	Someone produce a report	誰かが報告書を作成する	agt(produce(icl>manufacture(agt>thing,obj>thing)):08.@entry, someone:00.@topic)
			obj(produce(icl>manufacture(agt>thing,obj>thing)):08, report(icl>account) :0I)

Dog (icl>mammal)	Dog is a canine, mammal and animal. Dog can eat, whine, walk and die.	犬は犬、哺乳類や動物です。犬は、食べて駄々をこねる、歩いて、死ぬことができます。	aoj(:01.@entry,dog(icl>mammal):00)
			and:01(animal(icl>living thing):0S.@entry, mammal(icl>animal):0H)
			and:01(mammal(icl>animal):0H,canine(icl>tooth):09.@indef)

Using UNL expressions and UNL DeConverter it is possible to generate the explanation in other languages as well. However, the quality of the explanation depends on the quality of that language DeConverter. Therefore precision of the system highly relies on UNL DeConverter and the semantic background provided by UNL Ontology. As the users of this system are the editors of UNL Ontology, it helps them to improve the quality of manually built UNL ontology. The UNL dictionary builders can also differentiate the UWs from the natural language explanation without understanding the UNL language.

In this research we proposed the way to auto generate the meaning of each UWs from UNL Ontology. However, UNL Ontology by nature is a growing resource with millions of UWs. As UWs are not always understandable by human, the explanatory sentences are needed to develop necessary UWs for every language. Using our proposed solution computer can auto generate the meaning of UWS in different natural languages using the information provided in UNL Ontology. This contribution is very useful for the UNL dictionary editors, as they can see the computer generated explanation for each UWs in their mother language. However, the mechanism depends on the quality of EnConverter and DeConverter. In the presented approach the explanation generation rules are manually developed by human. In future, we would like to compare WordNet and the generated explanations of UNL Ontology in our EBMT system.

Chapter 10

Conclusion

We proposed an EBMT system for low-resource language using CSTs in the example-base. Our EBMT system is effective for low resource language like Bengali. We used WordNet to translate the unknown words which are not directly available in the dictionary. To translate an English sentence, it is first parsed into chunks. Then the chunks matched with the CSTs to find translation candidates. Then the system de-termines translation candidates for the identified unknown words from WordNet. Finally using generation rules the target-language strings has been produced.

CSTs improved the wide-coverage by 57 points and quality by 48.81 points in human evaluation. Currently 64.29% of the test-set translations by the system were acceptable. The combined solutions of CSTs and unknown words generated 67.85% acceptable translations from the test-set. Unknown words mechanism improved translation quality by 3.56 points in human evaluation.

Currently we used a small parallel corpus to generate CSTs. However to increase the performance we need a balanced parallel corpus (Salam et al. 2010). Although current system works well for small parallel corpus, the performance can decrease with larger parallel corpus. Because it will have many candidate CSTs. In future, we want to improve current CSTs selection mechanism. We plan to use statistical language model for future improvement. It can improve the generation quality.

In future, we would like to compare WordNet and the generated explanations of UNL Ontology in our EBMT system. In future we also want to evaluate the system using BLEU and other standard Machine Translation evaluation metrics.

References

[1]. MACHINE TRANSLATION: An Introductory Guide , *By Doug Arnold, Lorna Balkan, Siety Meijer, R.Lee Humphreys, Louisa Sadler*

[2]. A Statistical MT Tutorial Workbook, *Kevin Knight*

[3]. MTTK: An Alignment Toolkit for Statistical Machine Translation, By Yonggang Deng

[4]. Integrating Knowledge Bases and Statistics in MT By *Kevin Knight, Ishwar Chander, Matthew Haines,*

[5]. What's in a translation rule? *By Michel Galley, Mark Hopkins, Kevin Knight and Daniel Marcu*

[6]. Methods for Statistical Machine Translation for English-Arabic, By Tom Ledbetter

[7]. Verb Transfer For English To Urdu Machine Translation by Nayyara Karamat

[8]. Statistical Machine Translation with a Small Amount of Bilingual Training Data, By Maja Popovi´c, Hermann Ney

[9]. A New Decoding Algorithm for Statistical Machine Translation: Design and Implementation, By *Tanveer A. Faruquie Hemanta K. Maji, Raghavendra Udupa U.*

[10]. A Trainable Transfer-based Machine Translation Approach for Languages with Limited Resources By Alon Lavie, Katharina Probst, Erik Peterson, Stephan Vogel, Lori Levin, Ariadna Font-Llitjos and Jaime Carbonell

[11]. Contributions To English To Hindi Machine Translation Using Example-Based Approach, Phd Theses, Deepa Gupta.

[12]. D. Gupta and N. Chatterje., Study of Divergence for Example-Based English-Hindi Machine Translation. STRANS-2001, IIT Kanpur, 2001 pp. 43-51.

[13]. Balanced Bengali Language Corpus: A Proposal, *By Khan Md. Anwarus Salam, S M Murtoza Habib and Dr. Mumit Khan*, Research work in BRAC University in 2008.

[14]. H.A. Guvenir and I. Cicekli., Learning Translation Templates from Examples. Elsevier Science Ltd., 1998

[15]. R. Jain , R.M.K Sinha and A. Jain., ANUBHATRI: Using Hybrid Example-Based Approach for Machine Translation.. STRANS-2001, IIT Kanpur, 2001 pp. 20-32.

[16]. Verb Transfer For English To Urdu Machine Translation, Independent Study by Nayyara Karamat, FAST-Lahore, 2006

[17]. An Optimal Way Towards Machine Translation from English to Bengali, By Sajib Dasgupta, Abu Wasif and Sharmin Azam. In the Proceedings of the 7th International Conference on Computer and Information Technology (ICCIT), Bangladesh, 2004.

[18]. Shah Asaduzzaman and Muhammad Masroor Ali, "Transfer Machine Translation-An Experience with Bangla English Machine Translation System". In the Proceedings of the International Conference on Computer and Information Technology (ICCIT), Bangladesh, 2003.

[19]. "Example Based English to Bengali Machine Translation" B.Sc. Thesis of Khan Md. Anwarus Salam completed in August 2009, BRAC University

[20]. Arnold, D. et al., editors (1993). Special issue on Evaluation of MT Systems. Machine Translation 8 (1-2): 1-126.

[21]. Contributions To English To Hindi Machine Translation Using Example-Based Approach, Phd Theses in January 2005, Deepa Gupta, IIT Delhi.

[22]. D. Gupta and N. Chatterje., Study of Divergence for Example Based English-Hindi Machine Translation. STRANS-2001, IIT Kanpur, 2001 pp. 43-51.

[23]. Balanced Bengali Language Corpus: A Proposal, *By Khan Md. Anwarus Salam, S M Murtoza Habib and Dr. Mumit Khan*, Research work in BRAC University in 2008.

[24]. H.A. Guvenir and I. Cicekli., Learning Translation Templates from Examples. Elsevier Science Ltd., 1998

[25]. R. Jain, R.M.K Sinha and A. Jain., ANUBHATRI: Using Hybrid Example-Based Approach for Machine Translation.. STRANS-2001, IIT Kanpur, 2001 pp. 20-32.

[26]. Verb Transfer For English To Urdu Machine Translation, Thesis by Nayyara Karamat, FAST-Lahore, 2006

[27]. An Optimal Way Towards Machine Translation from English to Bengali, By Sajib Dasgupta, Abu Wasif and Sharmin Azam. In the Proceedings of the 7th International Conference on Computer and Information Technology (ICCIT), Bangladesh, 2004.

[28]. Makoto Nagao (1984). "A framework of a mechanical translation between Japanese and English by analogy principle". In A. Elithorn and R. Banerji. Artificial and Human Intelligence. Elsevier Science Publishers.

[29]. Abney, Steven. 1991. Parsing by chunks. In Principle- Based Parsing, pages 257–278. Kluwer Academic Publishers.

[30]. Diganta Saha, Sivaji Bandyopadhyay. 2006. *A Semantics-based English-Bengali EBMT System for translating News Headlines.* Proceedings of the MT Summit X, Second workshop on Example-Based Machine Translation Programme.

[31]. Diganta Saha, Sudip Kumar Naskar, Sivaji Bandyopadhyay. 2005. *A Semantics-based English-Bengali EBMT System for translating News Headlines,* MT Xummit X.

[32]. George A. Miller (1995). *WordNet: A Lexical Database for English.* Communications of the ACM Vol. 38, No. 11: 39-41.

[33]. Jae Dong Kim, Ralf D. Brown, Jaime G. Carbonell. 2010. *Chunk-Based EBMT.* EAMT, St Raphael, France.

[34]. Khan Md. Anwarus Salam, Mumit Khan and Tetsuro Nishino. 2009. *Example Based English-Bengali Machine Translation Using WordNetWordNet*. TriSAI, Tokyo.

[35]. Khan Md. Anwarus Salam, Yamada Setsuo and Tetsuro Nishino. 2010. *English-Bengali Parallel Corpus: A Proposal*. TriSAI, Beijing.

[36]. Md. Zahurul Islam, Jörg Tiedemann & Andreas Eisele. 2010. *English to Bangla phrase-based machine translation*. Proceedings of the 14th Annual conference of the European Association for Machine Translation.

[37]. R. Gangadharaiah, R. D. Brown, and J. G. Carbonell. Phrasal equivalence classes for generalized corpus-based machine translation. In Alexander Gelbukh, editor, Computational Linguistics and Intelligent Text Processing, volume 6609 of Lecture Notes inComputer Science, pages 13–28. Springer Berlin / Heidelberg, 2011.

[38]. Sajib Dasgupta, Abu Wasif and Sharmin Azam. 2004. An Optimal Way Towards Machine Translation from English to Bengali, Proceedings of the 7th International Conference on Computer and Information Technology (ICCIT).

[39]. Sudip Kumar Naskar, Sivaji Bandyopadhyay. 2006a. *A Phrasal EBMT System for Translating English to Bengali.* Workshop on Language, Artificial Intelligence and Computer Science for Natural Language Processing applications (LAICS-NLP).

[40]. Zhanyi Liu, Haifeng Wang And Hua Wu. *2006. Example-Based Machine Translation Based on Tree-string Correspondence and Statistical Generation.* Machine Translation, 20(1): 25-41

[41]. W.John Hutchins and Harold L. Somers. 1992. *An introduction to machine translation.* Academic Press.

[42]. Ying (Joy) Zhang, Nguyen Bach, Nguyen Bach. *Virtual Babel: Towards Context-Aware Machine Translation in Virtual Worlds.* Machine Translation Summit, MT Summit XII 2009, Ottawa, Canada, 01. August 2009

APPENDIX: SAMPLE TRANSLATIONS PRODUCED BY THE EBMT SYSTEM

#	**English**	**EBMT+CSTs+Unknown Words**
1.	Are you looking for an aardvark?	আপনি কি আর্ডভার্ক, এক ধরনের পশু খুঁজছেন?(A)
2.	This dog is really cool.	ডগ, এক ধরনের পশু আসলেই দারুন (A)
3.	WordNet is a..	শব্দজাল হচ্ছে.. (A)
4.	Sublexical units of a word	শব্দের উপ–আভিধানিক অংশ (A)
5.	This is a bluebird	এটা নীলপাখি .. (A)
6.	What is abstriction?	এ্যাবস্ট্রিকশান কি? (B)
7.	I am eating onigiri	আমি অনিগিরি খাচ্ছি(A)
8.	His name is Rupok.	তার নাম রুপক।(A)
9.	Japanese is the native language of around 120 million people worldwide	বিশ্বব্যাপী জাপানি হচ্ছে প্রায় ১২০ মিলিয়ন মানুষ –এর মাতৃভাষা (A)
10.	He is unfriendly	সে বন্ধুত্বপূর্ণ না (A)

List of Publications

Book Chapters

○**Khan Md. Anwarus Salam**, Hiroshi Uchida, Setsuo Yamada and Tetsuro Nishino. "Web Based UNL Graph Editor", Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing Studies in Computational Intelligence Volume 492, 2013, pp 219-228. (Book Chapter)

Refereed Journal Papers

○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, "Translation of Unknown Words for English to Bangla Machine Translation Using Transliteration", Special Issue on Best Papers of ICCIT 2011 and ICAEE 2011, JOURNAL OF MULTIMEDIA, VOL. 7, NO. 5, OCTOBER 2012 (peer-reviewed).

○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, UNL Ontology Visualization for Web, Journal of Convergence Information Technology. July 2013. (ISSN 1975-9320) (on press, peer-reviewed).

Best Paper Award

[1] Best paper award in SNPD, July 1 - 3, 2013, Honolulu, Hawaii, U.S.A.

[2] Best paper award in ICCIT, 2011, Dhaka, Bangladesh.

[3] Best presentation award in 13th APNG Camp, Seoul, Korea. August 2012.

International Conferences

1. ○**Khan Md. Anwarus Salam**, Hiroshi Uchida, Setsuo Yamada and Tetsuro Nishino. "Universal Words Relationship Question-Answering from UNL Ontology", 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel / Distributed Computing (SNPD) July 1 - 3, 2013 Honolulu, Hawaii, U.S.A. (oral presentation, peer-reviewed, received best paper award).
2. ○**Khan Md. Anwarus Salam**, Hiroshi Uchida, Setsuo Yamada and Tetsuro Nishino. "Universal Words Relationship Question-Answering from UNL Ontology", 2nd IEEE/ACIS International Conference on Computer and Information Science(ICIS 2013). June16-20, 2013,Toki Messe, Niigata, Japan (accepted-oral presentation, peer-reviewed).
3. ○Hiroshi Uchida, Meiying Zhu, **Md. Anwarus Salam Khan**, "UNL Explorer". 24th International Conference on Computational Linguistics (COLING 2012), Mumbai, India. page 453-458 (Demos presentation, peer-reviewed)
4. ○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, "Sublexical Translations for Low-Resource Language", Machine Translation and Parsing in Indian Languages, Mumbai, India. 2012. ISSN: 2226-2105. IEEE Catalog No. CFP1244S-CDR. ISBN: 978-4673-1152-6. May 2012a. (oral).
5. ○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, "Bangla phonetic input method with foreign words handling", Second Workshop on Advances in Text Input Methods (WTIM 2), Mumbai, India. 2012.
6. ○**Khan Md. Anwarus Salam**, H. Uchida, T. Nishino. "Multilingual Universal Word Explanation

Generation from UNL Ontology", Cognitive Aspects of the Lexicon (CogaLex), Mumbai, India. 2012.

7. ○**Khan Md. Anwarus Salam**, "Breaking the Language Barrier using Universal Networking Language (UNL)", 13th APNG Camp, Seoul, Korea. August 2012. (oral, peer-reviewed, best presentation award).
8. ○**Khan Md. Anwarus Salam**, Hiroshi Uchida, Setsuo Yamada and Tetsuro Nishino. "UNL Ontology Visualization for Web", 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Computing (SNPD 2012) August 08 - 10, 2012.
9. ○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, "Developing the First Balanced Corpus for Bangla Language", International Conference on Informatics, Electronics & Vision (ICIEV2012), Dhaka, Bangladesh. ISSN: 2226-2105. IEEE Catalog No. CFP1244S-CDR. ISBN: 978-4673-1152-6. May 2012a. (oral presentation, peer-reviewed).
10. ○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, "Using WordNet to Handle the Out-Of-Vocabulary Problem in English to Bangla Machine Translation", Global WordNet Conference, Matsue, Japan, (Editors Christiane Fellbaum et. al., Tribun EU, Brno, 2012, ISBN 978-80-263-0244-5). Page 35-39. January 2012b. (oral presentation, peer-reviewed).
11. ○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino, "Translating Unknown Words Using WordNet and IPA-Based- Transliteration", ICCIT, Dhaka, Bangladesh. Page 481-486. December 2011a (oral presentation, peer-reviewed, best paper award).
12. ○**Khan Md. Anwarus Salam**, Hiroshi Uchida and Tetsuro Nishino. "How to Develop Universal Vocabularies Using Automatic Generation of the Meaning of Each Word", 7th International Conference on Natural Language Processing and Knowledge Engineering (NLPKE'11), Tokushima, Japan. ISBN: 978-1-61284-729-0. Page 243 - 246 November 2011b. (oral presentation, peer-reviewed).
13. ○**Khan Md. Anwarus Salam**, Setsuo Yamada and Tetsuro Nishino. "Example-Based Machine Translation for Low-Resource Language Using Chunk-String Templates", 13th Machine Translation Summit, Xiamen, China. September 2011c. (poster presentation, peer-reviewed).
14. ○**Khan Md. Anwarus Salam**, Yamada Setsuo and Tetsuro Nishino, "English-Bengali Parallel Corpus: A Proposal ", TriSAI - 2010, Beijing, China, October 2010a. (oral presentation).
15. ○**Khan Md. Anwarus Salam**, Tetsuro Nishino and Mumit Khan, "Example Based English-Bengali Machine Translation Using WordNet", TriSAI, Tokyo, Japan, October 2009. (oral presentation).
16. ○**Khan Md. Anwarus Salam**, Tetsuro Nishino. "EEQRA: Evolving Easy Quantifiable Reading Assistant", 13th APNG Camp, APRICOT 2011, Hong Kong, February 2011. (oral, peer-reviewed).
17. ○**Khan Md. Anwarus Salam**, Tetsuro Nishino. "Question and Answering System for Learning-Resource-Repository", Young Researchers' Roundtable 2010 on Spoken Dialog Systems, Tokyo (poster).
18. ○**Khan Md. Anwarus Salam**. "Position paper on Spoken Dialog Systems", Young Researchers' Roundtable on Spoken Dialog Systems 2011 (Organizing Committee member), Oregon, USA. June 2011. (poster presentation, peer-reviewed).