Translation Rules and ANN based model for English to Urdu Machine Translation

SHAHNAWAZ[1]
R. B. MISHRA[2]

IT-BHU, Institute of Technology, Banaras Hindu University
Department of Computer Engineering
Varanasi, U.P., India-221005
[1]shahnawaz.rs.cse@itbhu.ac.in
[2]ravibm@bhu.ac.in

Abstract. In this paper we discuss the working of our English to Urdu Machine Translation (MT) system. We used feed-forward back-propagation artificial neural network for the selection of Urdu words/tokens (such as verb, noun/pronoun etc.) and translation rules for grammar structure equivalent to English words/tokens and grammar structure rules respectively. As English is SVO class language while Urdu is SOV class language so grammar structure transfer is main task in English-Urdu machine translation problem. Our system is able to translate sentences having gerund, having infinitives (maximum two), having prepositions and prepositional objects (maximum three), direct object, indirect object etc. Neural network works as the knowledge base for linguistic rules and bilingual dictionary. Bilingual dictionary not only stores the meaning of English word in Urdu but also stores linguistic features attached to the word. The output of our system is presented in Romanized Urdu. The n-gram blue score achieved by the system is 0.6954; METEOR score achieved is 0.8583 and F-score of 0.8650.

Keywords: Neural network, back-propagation, rule based translation, English, Urdu, machine translation system, Artificial Intelligence

(Received June 4th, 2011 / Accepted September 15th, 2011)

1 Introduction

Machine translation, also referred as MT, is the process of translating one natural language (as English) text to another natural language (as Urdu) text by the use of computing machine. Machine Translation is an automated process in which translation job is done by the Computer Software. Machine Translation is an application of computer linguistic. Computer linguistic is an interdisciplinary field of computer science and requires language and computer experts. Translation as an art of rendering a work of one language into another is as old as written literature [1]. As the needs of multilingual information are increasing in business, industries and economics, machine translation cannot be ignored. If MT researchers are able to develop a perfect multilingual machine translation system, people with different languages can share ideas and information worldwide on every topic as research, political, business, economical, and socio-cultural etc. The purpose of a translation process whether machine translation or human translation is that meaning of the text being translated should not change. There are many different machine translation system available online as well as desktop systems. G.R. Tahir, S. Asghar and N. Masood in [21] analyze the results from most popular MT systems like Babylon 8, World-lingo, PakTranslations, ApniUrdu and MT by FAST-NU and find out that the translation result is ambiguous and have wrong sense of meaning. Many languages spoken in the developing countries have been ignored by the researchers though these languages are spoken by a large population [7]; it holds the same status for Urdu language also.

Urdu language is one of the languages from the family of Indo-Aryan languages. There are some 210 languages and dialects in the family of Indo-Aryan languages [17]. Urdu is closely related to Hindi with a similar grammatical structure, but differences in script and vocabulary [2]. Hindi and Urdu are sister languages having many common linguistic features [6]. They are structurally very close to each other and use similar postpositions, verb morphology as well as complex predicate verb structure. There are between 60 and 70 million self-identified native speakers of Urdu in different continents. English has its own importance with respect to international language; and is a knowledge containing language [3]. A good translator will remove this gap and people will be able to communicate without any language barrier.

1.1 Literature survey

The machine translation work in English to Urdu machine translation is substantially lagging in spite of large population of Urdu speakers. The English to Urdu machine translation system developed by Tafseer Ahmed, Sadaf Alvi in [19] uses transfer based approach and bottom up chart parsing for English-Urdu translation task. An expert system based English-Urdu machine translation work in [11] relies on QTAG for part of speech tagging and uses knowledge base for grammatical patterns and gender aware dictionary. Maryam Zafar et al in [25] developed an interactive machine translation system using Example based approach. This system uses Levenshtein algorithm and semantic distance algorithm for searching bilingual corpus. System uses n-ary product for listing all the possible translations for the input sentence. System has rules for ordering the translated text and also supports homograph, idioms and some other linguistic features. The work discussed in [10] is a bidirectional English-Urdu machine translation system with natural language processing. This system uses rule based methodology with bottom up parsing and dynamic dictionary for translation. AGHAZ [13] is an Expert System based automatic translator for English to Urdu machine translation. It has an expert system, patterns or rules and a rich knowledge base which stores English words with their Urdu meaning, part of speech, gender, number and multi-word information. In a work of English to Urdu translation, R M K Sinha in [18] uses his English-Hindi MT System as a model to translate Urdu from English. This English-Hindi MT system is built by using rule based approach and a pseudo Interlingua. In this work, a Hindi-Urdu mapping table is used which stores Urdu meaning of Hindi words and information that affect the composition of Urdu text; to generate Urdu text from the Hindi output of this system. Sampark in [22] is a machine translation system for automated machine translation among Indian languages including Urdu, developed by the Consortium of institutions include IIIT Hyderabad, University of Hyderabad, KBC, Chennai, IIT Kharagpur, CDAC (Noida,Pune), Anna University, IIT Kanpur, IISc Bangalore, IIIT Alahabad, Jadavpur University, Tamil University [22]. This system uses hybrid methodology which consists of rule based approach and dictionaries and statistical machine learning techniques. A proposed knowledge based machine translation system in [21] is an enhancement of Sampark model which considers most of the types of ambiguities and uses text mining and data mining techniques for machine translation. A brief overview of English-Urdu machine translation works discussed here is given below table (to see Table 1).

Our English to Urdu machine translation system works at paragraph level as well as for a single sentence. When a source language text is entered in the system as input, system processes the text into sentences. Then each sentence is translated and rearranged to generate the Urdu translation. First of all, contraction removal module removes contraction from all the sentences. Then each sentence is parsed and tagged. The output of parser and tagger are processed to extract all the information related to each word present in the sentence and now each word is transformed into an object which contains information (like part of speech, dependency, word position in the sentence etc.) about this word and sentence is transformed into a group of knowledgeable objects. Now these objects are given to the grammar analysis and sentence structure recognition module which processes the information and recognizes the grammar tokens (like subject, object, verb, infinitive, gerund etc.) of the sentence and generate the grammatical structure using the rule base. Artificial Neural Network (ANN) and Rule based sentence structure mapping module maps this grammatical structure to corresponding Urdu grammar structure and ANN based Urdu word mapping module maps each word from the each sentence part to the Urdu word. Each part is now arranged according to the Urdu grammatical structure obtained from ANN and Rule based sentence structure mapping module. Syntax addition module adds verb marker and case markers based on the information attached with Urdu words and in knowledgeable objects. Translation of each sentence is generated and presented in Romanized Urdu.

Table 1: Overview of English-Urdu machine Translation Works / Systems

S. No.	System/ Work(Year)	ResearchTeam	Work/System Methodology
1.	English to Urdu Translation System (2002)	Tafseer Ahmed, Sadaf Alvi	Transfer based approach, Bottom Up Chart Parsing
2.	Expert system driven approach to generate natural language (2003)	Expert system driven approach to generate natural language (2003)	Expert system based approach, QTAG, knowledge base for grammatical patterns and gender aware dictionary
3.	Urdu Translation Engine (2004)	Mohammad Kashif Shaikh, Hussain Hyder Ali Khowaja, Muzammil Ahmed Khan	Natural language processing, rule based, bottom up parsing and dynamic dictionary Bidirectional English-Urdu machine translation
4.	AGHAZ (2005)	Uzair Muhammad, Kashif Bilal, Atif Khan, and M. Nasir Khan	Expert System based approach, knowledge base for grammatical patterns and gender aware dictionary, Handles multiple words and proper noun
5.	Interactive English-Urdu machine translation (2009)	Maryam Zafar et al	Example based approach, Levenshtein and semantic distance algorithm, is N-ary Product, ordering rules Supports homograph, idioms and some other features.
6.	English-Urdu Machine Translation Via Hindi (2009)	R. M. K. Sinha	Mapping of Hindi output from English-Hindi MT system which is based upon PLIL (pseudo Lingua for Indian Languages) and Rule based approach.
7.	Sampark (2009)	Sampark machine translation Team-Consortium of Institutions	Sampark machine translation Team-Consortium of Institutions
8.	Knowledge based Machine Translation System (2010)	Ghulam Rasool Tahir, Sohail Asghar, Nayyer Masood	Text mining and Data mining techniques; Focuses on adding semantics.

The further work discussed in this paper is divided into the following sections: The second section of this paper gives a brief overview of linguistic characteristics of Urdu language. Third section comprises the discussion about our proposed work, our system architecture and description. This section also discusses encoding-decoding and neural network module. Software implementation of the system and working of our system is explained in the fourth section. Then we discuss results and evaluation of our system. Last section is conclusion and future work.

2 Linguistic characteristics of Urdu

2.1 Origin and Vocabulary

Among all the languages in the world, Urdu is most closely similar to Hindi. Both the languages Hindi and Urdu have originated from the Delhi region dialect and other than the minute details, these languages share their morphology. Hindi language has adopted many words from Sanskrit while Urdu language has borrowed a large number of its vocabulary items from Persian and Arabic. Urdu has also borrowed words from Turkish, Portuguese and English [5]. There are a large number of words which have found a place in Urdu Language, from the Persian; have differently nuanced connotations and usages [14].

2.2 Grammar Structure

One of the most significant aspects of Urdu language grammar structure is its word order which is SOV (subject, object, and verb). This order does exhibit some flexibility as the subject pronouns are frequently dropped.

2.3 Nouns

Nouns in Urdu Language grammar have two types of gender (masculine/feminine), two type of numbers (singular/plural) and three cases (vocative, direct and oblique). All nouns in Urdu, when used within a sentence, will be inflected for number and case. Suffix specifies gender on verbs and adjectives in [5]. example: pagal → pagalpan (madness), ghabrana → ghabrahat (anxiety) Common suffixes can be used to drive nouns from other words. These forms are masculine and feminine nouns. One notable point is that the borrowed Arabic and Persian plurals form of the noun are never inflected in Urdu [14].

2.4 Verbs

Verbs in Urdu language have two nonfinite forms, root and infinitive. The infinitives comprise a verbal stem and a suffix. The stem may itself comprise verbal root and suffix. e.g. ana (to come), jana (to go). In this example a- and ja- are the root and -na is suffix. The infinitive forms of all verbs are marked masculine from grammatical point of view. They neither occur in the plural, nor in the vocative. There are many verbal forms of stem having different endings added to them to form the patterns of different verb forms. In Urdu, subjunctive is the finite verbal form which conveys the week conjectures on the Urdu part of speaker [14]. Another two verbal forms that may be finite or nonfinite are the perfective and imperfective participles. The imperfective participle ends with -ta,-te,-ti,-tin. In case of perfective participles, participle ends with e.g -a,-e,-i,-in. But the situation will be different whenever there is any verbal stem that ends in a vowel; we have to add a/y before masculine singular ending [8, 4]. The future verb forms in Urdu cannot be decided from the verb stems rather it is decided from subjunctive forms. The endings for the future forms are (e.g. -ga,-ge,-gi) [14]. Common use of semi auxiliary elements also gives various semantic connotations [5].

2.5 Post-positions

In Urdu, post positions take place after noun phrase head which is an absolute contrast to English language where a variety of elements occur between preposition and the governed noun. This process has helped to reach a conclusion that Urdu has a lot of diversity of cases [5] e.g. genitive, accusative etc.

3 Our proposed work

In our work of English to Urdu machine translation, we used a neural network and rule based approach. Rule based approach is the classical approach of machine translation. In rule based machine translation approach, system is fed with linguistic rules and bilingual dictionaries. System parses and analyses the grammatical structure of the source language text, this structure is then transformed to the target language structure with the help of the linguistic rules. When the structure is transformed, target language text is generated by the use of bilingual dictionaries and linguistic rules. Many systems have been developed using rule based machine translation, in which main systems are as Systran, Eurotra and Japanese MT System. Neural networks are a possible solution to the machine translation problem. Neural networks have the ability of learning by examples. Neural network has proven very useful in various natural language processing tasks [8]. PARSEC [9], JANUS [24] and English-Sanskrit MT system [12] use neural network approach for natural language processing task and automated machine translation. Our English to Urdu machine translation system uses Feed-Forward Back-Propagation Neural Network with rule based machine translation approach. Neural networks are very efficient in pattern matching. Machine translation using rule based approach is consistent and of predictable quality. Our system uses neural network as knowledge base for storing linguistic rules and also as knowledge bilingual dictionary. Neural network maps Urdu words/tokens (such as verb, noun/pronoun etc.) and grammar structure rules equivalent to English words/tokens and grammar structure rules. These words/tokens are then processed, rules are interpreted and all the parts of the sentence are arranged according to the interpreted rules.

3.1 System Architecture and Description

The block diagram of our English to Urdu Machine Translation System is shown in figure (to see Figure1). There are eight main modules such as Contractions Removal, Parser and Tagger, Knowledge Extraction, Grammar Analysis and Sentence Structure Recognition, ANN (Artificial Neural Network) and Rule Based Sentence Structure Mapping, ANN Based Urdu word Mapping, Rule Based Syntax Addition and Urdu Sentence Generation in our System. Working of each module is explained below.

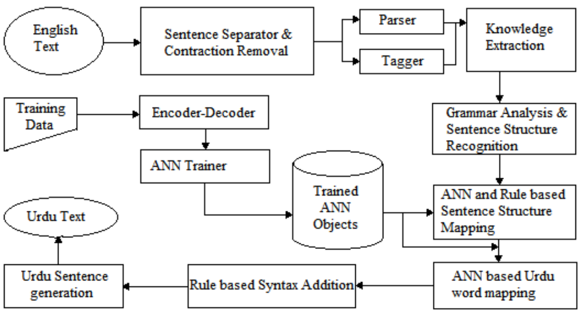

English Text
Sentence Separator & Contraction Removal
Parser
Tagger
Knowledge Extraction
Training Data
Encoder-Decoder
Grammar Analysis & Sentence Structure Recognition
ANN Trainer
Trained ANN Objects
ANN and Rule based Sentence Structure Mapping
Urdu Text
Urdu Sentence generation
Rule based Syntax Addition
ANN based Urdu word mapping

Figure 1: System Architecture

Sentence Separator and Contractions Removal: The translation process starts with English text input to this module. This module first separates the paragraph into sentences. Then each sentence is processed. If any contraction is present in the sentence it is removed. Then it is passed to the parser and tagger module. Contractions are common in spoken English and now becoming informal in written English also. In this step, we replace contractions with their respective full form. For example, I'm, you've, she'll and they‘d etc will be replaced by respectively I am, you have, she will and they had/would etc. Similarly, negative contractions aren't, needn't, won‘t etc will be replaced by respectively are not, need not, will not etc.

Parser and Tagger: Processed text from Contraction Removal module is given as input to the Parser and Tagger module. Stanford typed dependency parser is used for parsing the English Text. Stanford parser is

the implementation of probabilistic natural language parsers, both highly optimized PCFG and lexicalized dependency parsers, and a lexicalized PCFG parser. Probabilistic parsers use knowledge of language gained from hand-parsed sentences to try to produce the most likely analysis of new sentences. These statistical parsers still make some mistakes, but commonly work rather well [16]. The parser provides Stanford typed dependencies as output. The output of parser for an English sentence is shown below:
Students must tell the result to their parents.
$nsubj(tell-3, Students-1), aux(tell-3, must-2), det(result-5, the-4), dobj(tell-3, result-5), poss(parents-8, their-7), prep_to(tell-3, parents-8)$.
We are using Stanford POS tagger for tagging the English text. A Part-Of-Speech Tagger (POS Tagger) assigns parts of speech to each word (and other token), such as noun, verb, adjective, etc. The Stanford POS tagger uses the Penn Treebank tag set and is implemented using maximum entropy tagging algorithm [20]. POS tagger adds part of speech information to each word (and other tokens) in the text. The output of tagger for an English sentence is shown below:
Students must tell the result to their parents.
Students/NNS must/MD tell/VB the/DT result/NN to/TO their/PRP$ parents/NNS.

Knowledge Extraction: The function of this module is to process the typed dependency obtained from parser and to process tagged text from tagger. This module extracts information from parser and tagger for each part of the sentence. Each part of the sentence is converted to knowledgeable object by adding all the information associated with it and sentence is represented as a collection of knowledgeable objects.

Grammar Analysis and Sentence Structure Recognition: This module processes the collection knowledgeable objects and recognizes parts of the sentence e.g. subject, main verb, auxiliary verb, object, indirect object etc. Tense of the sentence is recognized with the help of main verb and auxiliary verb. Sentence voice whether it's passive or active, is also recognized in this phase. Sentence type is detected from the knowledge present in the collection of knowledgeable objects. On the basis of knowledge obtained, sentence parts and attributes (tense, voice, type etc.) are analyzed and sentence grammatical structure is generated with the help of rule base.

ANN and Rule based sentence structure mapping: Generated grammatical structure and attributes are passed to ANN and Rule based sentence structure mapping module. This module gathers information and makes a query to obtain the corresponding grammar structure for target languages i.e. Urdu. This query is coded into numeric form (decimal number). Artificial Neural Network (ANN) is trained on a data set of decimal encoded Rule Base for English Grammar and corresponding Urdu Grammar in which, various parts of the structure are separated by space. On query, ANN model returns Urdu grammar structure corresponding to the attribute and English grammatical structure knowledge encoded in the query. The returned structure is also in the numeric form which is then decoded to textual form for further processing.

ANN Based Urdu word mapping: Sentence parts (words or tokens) are transformed according to the Urdu grammar structure obtained from last module. Now each sentence part has to be translated. Urdu word mapping module encodes each word into numeric form and looks for each word in the bilingual ANN model which is trained for word mapping, and gets the corresponding English word and associated information in numeric form. This result is decoded to textual form which contains Urdu meaning of the word and coupled information. For a word, which is noun/pronoun, coupled information will be the number, person and gender and a word which is verb, coupled information will be its weak verb. Verbs in Neural Network model are trained with their base form meaning and weak verb if there is any.

3.2 Translation Rules

Translation Rules have been created for various classes of the sentences. Our system is able to handle all forms (affirmative, negative and interrogative) of the English simple sentences. Syntax addition to verb and case marker addition to subject and object will be added on the basis of information of tense, subject and object gender, number and person. For example for the following sentence: English Sentence: I lent my pen to a friend. Following translation rule will be used: IF (Sentence structure is SVOPPO and tense is Past-Indefinite and sentence is affirmative in active voice) THEN (Urdu grammar = subject (S) + object (O) + prepositional object (PO) + preposition (P) + verb (V)). Syntax addition: As direct object is present in the sentence so case marker 'ne' has to be added and marker 'a' to verb will also be added. This is decided on the basis of tense, sentence structure and coupled information (number, person, gender) with the Urdu meaning of the word. Syntax addition rules have been written for each tense considering all cases of number, gender, and person and sentence structure. The general structure for the grammar rules for training neural

network as follows
I/p → gclass + tense + type + category + voice.
O/p → urdu grammar
E.g. I/p → svo + pastInd + s + aff + act O/p → sov.

Where gclass is the grammar class of sentence like SVO, tense like Past Indefinite, type of the sentence is simple, complex, imperative etc, category is affirmative, interrogative etc and voice is active or passive. Translation rules for the following structures of the sentences have been written SVSc, SV, SVO, SVIoO, SVIn, SVInIn, SVInO, SVG, SVGO, SVpPO, SVpPOpPO, SVpPOpPOpPO, SVOpPO, SVOpPOpPO, SVOpPOpPOpPO; where S = Subject, V = Verb, Sc = Subject Compliment, Io = Indirect Object, In = Infinitive, G = Gerund, p = preposition and PO = Prepositional Object. Some examples of translation rules as follows:

English Sentence (E.S.): Mr S Khan is a research scholar
IF (sentence structure is SVSc and tense is present and affirmative sentence in active voice)
THEN (Urdu grammar = S + Sc + V)
E.S.: Has the bell rung?
IF (sentence structure is SV and tense is present perfect and verb interrogative sentence in active voice)
THEN (Urdu grammar = kya + S + V)
E.S.: The boy hadn't lost his pen.
IF (sentence structure is SVO and tense is past perfect and negative sentence in active voice)
THEN (Urdu grammar = S + O + negative word + V)
E.S.: Why does he not want to go to watch the movie?
IF (sentence structure is SVInInO and tense is present Indefinite and interrogative-negative sentence in active voice)
THEN (Urdu grammar = S + O + In2 + question word + negation word + In1 + V).
E.S.: I lent my pen to my friend.
IF (sentence structure is SVOpPO and tense is past Indefinite and interrogative-negative sentence in active voice)
THEN (Urdu grammar = S + O + PO + p + V).

3.3 Encoder-Decoder

We created a data set of input-output pairs of English-Urdu words with associated knowledge and another data set of input-output pairs of grammar rules. Encoder-Decoder converts this training data into numeric coded form which is suitable to be used as input for the ANN trainer. Each English alphabet is represented as a five bit binary number (a = 00001, b = 00002 and so on)(to see Table 2). Value of each alphabet is converted to decimal by dividing 26 (a = 1/26, b= 2/26 and so on) to train the neural network. Some special characters are also used for correct representation of a word in Roman Urdu. All the special characters are assigned values higher than one. For training neural network, we encode each character of the words/tokens and grammar structure to numeric form as explained above.

Table 2: English Alphabet Encoding

$S.No.$	$Alphabet$	$5 - bit$ $binary$	$Decimalcode$ $forthe$ $alphabet$ $(binary/26)$
1.	a	00001	0.038462
2.	b	00010	0.076923
3.	c	00011	0.115385
4.	d	00100	0.153846
5.	e	00101	0.192308
6.	f	00110	0.230769
7.	g	00111	0.269231
8.	h	01000	0.307692
9.	i	01001	0.346154
10.	j	01010	0.384615
11.	k	01011	0.423077
12.	l	01100	0.461538
13.	m	01101	0.500000
14.	n	01110	0.538462
15.	o	01111	0.576923
16.	p	10000	0.615385
17.	q	10001	0.653846
18.	r	10010	0.692308
19.	s	10011	0.730769
20.	t	10100	0.769231
21.	u	10101	0.807692
22.	v	10110	0.846154
23.	w	10111	0.884615
24.	x	11000	0.923077
25.	y	11001	0.961538
26.	z	11010	1.000000
27.	a (bar)	11011	1.038461
28.	e (bar)	11100	1.076923
29.	i (bar)	11101	1.115384
30.	n (acute)	11110	1.153846
31.	u (bar)	11111	1.192307
32.	Space	00000	0

3.4 Neural Network and Training

We have created a two-layer feed-forward neural network. First layer in this network is sigmoid and second layer is linear. We trained this network with Levenberg-Marquardt algorithm. There are many numerical optimization techniques to speed up the convergence of back-propagation algorithm. Different algorithms perform differently for a given problem. The results, presented in [4], show that Levenberg-Marquardt algorithm is very efficient for training the networks having up to a few hundred weights. We have trained neural network for grammar structure rules with a data set of around 465 input-output pair of grammar rules. The input layer of grammatical structure network contains 42 nodes, hidden layer contains 100 nodes and output layer contains 30 nodes. Mean squared error goal was set to training error of 10^{-8} which was achieved after 29 epochs. The neural network for knowledgeable bilingual dictionary has been trained with a data set of around 9000 input-output pair of English-Urdu words with associated knowledge. The input layer of bilingual dictionary network contains 10 nodes, hidden layer contains 100 nodes and output layer contains 32 nodes (for meaning and other information). Mean squared error goal was set to training error of 10^{-8} which was achieved after 333 epochs.

3.4.1 ANN Based Mapping Process

We used feed-forward back-propagation artificial neural network for the selection of Urdu words/tokens (such as verb, noun/pronoun etc) and grammar structure rules equivalent to English words/tokens and grammar structure rules. There are three main steps in mapping process as follows:

1) Encoding of English words/tokens or grammar structure to numeric code.

2) Mapping of English numeric code: Data sets are fed to Neural Network from which ANN selects the Urdu equivalent of the English words/tokens or grammar structure provided for Translation.

3) Decoding the code of the obtained Urdu words/tokens or grammar structure.

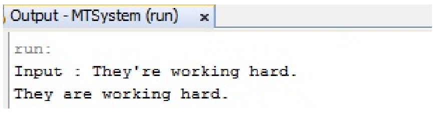
Once we get the equivalent words/tokens or grammar structure, Urdu meaning and information is extracted and processed.

4 Implementation

We have implemented our English-Urdu machine translation system on java platform. We used java jdk1.5 version for its compatibility with Matlab 7.1. System is implemented in java except from the neural network module. Neural network model is trained, tested and successfully implemented using Matlab 7.1 neural network library. Neural network works as the knowledge base for linguistic rules and bilingual dictionary. Bilingual dictionary does not only store the meaning of English word in Urdu but also store linguistic knowledge (e.g. verb, noun, pronoun, number, person and gender etc) attached to the Urdu word. We trained the two-layer feed-forward neural network with Levenberg-Marquardt back-propagation algorithm. We created two separate neural networks one for grammar structure rules and one for English-Urdu bilingual knowledge dictionary as Urdu equivalent of English words also have associated knowledge about the word like verb, noun, pronoun, number, person and gender. The neural networks model for grammar structure rules gives the Urdu equivalent grammatical structure to English sentence being translated and the neural model for bilingual knowledgeable dictionary gives the Urdu equivalent word and associated knowledge about the word. A java class does coding and decoding of the tokens and linguistic rules and gives to the neural networks as input for mapping them to their equivalent Urdu tokens and linguistic rules. To automate the process we created a java class for creating training data in numeric form with help of coding and decoding java class from a text file where data is present in human readable form as a word in numeric form is difficult to read by a human but easy for a program. Neural network then maps these numeric values and produces equivalent result in numeric form which are then again passed to the java class which decodes numeric data and present in the string form. This knowledge is further processed and Urdu meaning and attached information is extracted. Suffix in the verb and marker with the subject are attached on the basis of knowledge obtained from the neural network and information obtained in the Grammar Analysis and Sentence Structure Recognition module. These parts are then arranged according to the grammar structure obtained from grammatical structure network and the output is presented in Romanized Urdu form.

5 Results and Evaluation

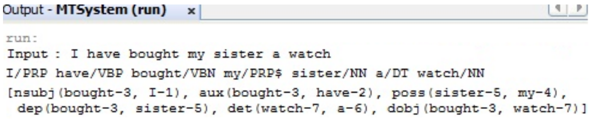
Our system is also able to handle contractions if present in the English input text figure 2 (to see Figure2).



```
run:
Input : They're working hard.
They are working hard.
```

Figure 2: Contraction Removal

The output of Contraction Removal module is passed to the Parser and Tagger module which uses Stanford parser and tagger for parsing and tagging the input English sentence. The output of the Parser and Tagger module is shown in the figure 3 (to see Figure3).



```
run:
Input : I have bought my sister a watch
I/PRP have/VBP bought/VBN my/PRP$ sister/NN a/DT watch/NN
[nsubj(bought-3, I-1), aux(bought-3, have-2), poss(sister-5, my-4),
 dep(bought-3, sister-5), det(watch-7, a-6), dobj(bought-3, watch-7)]
```

Figure 3: Parser and Tagger Output

Knowledge extraction module processes the result obtained from Parser and Tagger module and converts each part of the sentence to a knowledgeable object by adding all the information associated with it. Sentence is now represented as a collection of knowledgeable objects which are then given as input to the Grammar Analysis and Sentence Structure Recognition module. This module analyzes these objects and identifies the attributes of the grammar for the English sentence as tense, voice, sentence type, subject, main verb, auxiliary verb, object, indirect object etc. Now each token and grammatical structure is mapped from the neural network as explained earlier in the implementation section. Verbs meaning are stored in the base form so suffix has to be added according to the tense, gender and number of the subject or object sometimes which are appended on the basis of knowledge obtained from the neural network and tense of the sentence. Case marker words like ka, ke, ko, ne, ki etc also attached with the subject on the basis of knowledge obtained from the neural network and information obtained in the Grammar Analysis and Sentence Structure Recognition module. All the parts of the sentence are then arranged according to the grammar structure rule obtained from grammatical structure network and the output is presented in Romanized Urdu form as below:

Input English Text: Ram Kumar Singh is a student. He lives in Shimla. Shimla offers you refreshing environment. He enjoys playing cricket. He likes singing. He went to the market with his father. He saw an old man in the market. The old man was buying a book for his wife from the market. He bought a pen for his sister. He met his friends. They wanted to go to watch the movie. He decided to watch the movie.

Output Urdu Translation: RAM KUMAR SINGH ek talib-e-ilm hai | wah SHIMLA me rahta hai | SHIMLA tumko tazagi bhara mahaul deta hai | wah cricket khelna lutf uthata hai | wah gana pasand karta hai | wah apne walid ke sath bazar ko gaya tha | wah bazar me ek boodha adami dekha tha | boodha adami bazar se apni biwi ke liye ek kitaab kharid raha tha | wah apni bahan ke liye ek kalam kharida tha | wah apne doston mila tha | ve film dekhna jana chahate the | wah film dekhna faisla kiya tha |

The words which are not present in the bilingual dictionary are printed as it is in the translation in capitals.

5.1 Evaluation

The problem of evaluation is same as the problem of translation. Various methods have been employed for evaluating the quality of machine translation output. Some features can be evaluated automatically for example fluency can be checked by n-gram analysis of reference translations are available and some can't as meaning sense of translation. It is hard to compare between two different Machine Translation algorithms objectively.

The evaluation scores for twenty-eight randomly selected sentences [shown in table 4,5 (to see 4 and 5)] of various classes are shown in the table 3 (to see 3) below. In these tables 4 and 5 sentence mean English sentence, candidate is the translation output of our machine translation system and reference is the Urdu translation of the English sentence by a human expert.

Table 3: Calculated score for the sentences shown in table 4, 5

$S.No.$	$BLEU$	P	R	M	F
1	1.0000	1.0000	1.0000	0.9998	1.0000
2	1.0000	1.0000	1.0000	0.9993	1.0000
3	1.0000	1.0000	1.0000	0.9995	1.0000
4	0.8105	1.0000	1.0000	0.9995	1.0000
5	0.4544	0.5556	0.5556	0.5533	0.5556
6	0.4137	0.7500	0.7500	0.7361	0.7500
7	0.8092	0.9091	0.9091	0.9055	0.9091
8	0.6776	0.8889	0.8889	0.8880	0.8889
9	0.3119	0.7143	0.7143	0.6371	0.7143
10	0.3708	0.8333	0.8333	0.8067	0.8333
11	1.0000	1.0000	1.0000	0.9990	1.0000
12	0.5435	0.8571	0.8571	0.8552	0.8571
13	0.4208	0.8333	0.8333	0.8300	0.8333
14	1.0000	1.0000	1.0000	0.9985	1.0000
15	0.5630	0.8889	0.7273	0.7212	0.8000
16	0.5667	0.8333	0.8333	0.8300	0.8333
17	0.5667	0.8333	0.8333	0.8300	0.8333
18	1.0000	1.0000	1.0000	0.9960	1.0000
19	0.7140	0.8750	0.7778	0.7773	0.8235
20	0.8101	0.8571	0.8750	0.8630	0.8660
21	0.8914	1.0000	0.9000	0.9041	0.9474
22	0.2762	0.5714	0.5714	0.5670	0.5714
23	0.8914	1.0000	1.0000	0.9993	1.0000
24	1.0000	1.0000	1.0000	0.9977	1.0000
25	1.0000	1.0000	1.0000	0.9977	1.0000
26	1.0000	1.0000	1.0000	0.9922	1.0000
27	1.0000	1.0000	1.0000	0.9977	1.0000
28	1.0000	1.0000	1.0000	0.9985	1.0000

BLEU in [15] (Bilingual Evaluation Understudy) is an IBM-developed metric, uses a modified form of precision (modified n-gram precision) to compare the candidate translation against reference translations. It takes the geometric mean of modified precision scores of the test corpus and then multiplies the result by exponential brevity penalty factor to give the BLUE score. Modified precision score can be calculated as follows:

$$p_n = \frac{\Sigma_C \in \{_C andidates\} \Sigma_n - gram \in_C Count_c lip(n-gram)}{\Sigma_{\acute{C}} \in \{_C andidates\} \Sigma_n - g\acute{r}am \in_{\acute{C}} Count_c lip(n-g\acute{r}am)}$$

Where C is the set of candidate translation sentences and C' is the set of reference sentences. Count clip in

Table 4: Twenty-eight randomly selected sentences

$S.No$	$Sentence, CandidateandReference$
1.	Sentence: Why has he bought a watch for his sister from the market? Candidate: wah bazar se apni bahan ke liye ek ghadi kyon kharid chuka hai Reference : wah bazar se apni bahan ke liye ek ghadi kyon kharid chuka hai
2.	Reference : wah bazar se apni bahan ke liye ek ghadi kyon kharid chuka hai Candidate: ladaka SALESMAN se ek kitaab kyon kharid raha tha Reference: ladaka salesman se ek kitaab kyon kharid raha tha
3.	Sentence: The girl was singing a song with her friends. Candidate: ladaki apne doston ke sath ek gana ga rahi thi Reference: ladaki apne doston ke sath ek gana ga rahi thi
4.	Sentence: Why did the teacher give homework to us? Candidate: ustad hamko ko ghar ke liye kam kyon diya tha Reference: ustad ne hamko ghar ke liye kam kyon diya tha
5.	Sentence: I bought 10kg mango for my sister. Sentence: I bought 10kg mango for my sister. Sentence: I bought 10kg mango for my sister.
6.	Sentence: I lent my pen to a friend. Candidate: mai ek dost ko mera kalam diya tha Reference: maine ek dost ko apna kalam diya tha
7.	Sentence: Why did he not go to the market with his friends? Candidate: wah apni doston ke sath bazar ko kyon nahi jata hai Reference: wah apne doston ke sath bazar ko kyon nahi jata hai
8.	Sentence: He went to the market with his friends. Candidate: wah apni doston ke sath bazar ko gaya tha Reference: wah apni doston ke sath bazar ko gaya tha
9.	Sentence: These books belong to me. Candidate: yen kitaben mujh ko talluk rakhta hai Reference: yen kitaben mujh se talluk rakhti hai
10.	Sentence: I like reading books. Candidate: mai padh kitaben pasand karta hun Reference: mai kitaben padhna pasand karta hun
11.	Sentence: Has he finished working? Candidate: Kya wah kam karna khatm kar chuka hai Reference: Kya wah kam karna khatm kar chuka hai
12.	Sentence: Where does he want to go to see the movie? Candidate: wah film dekhna kahaan jana chahta hai Reference: wah film dekhne kahaan jana chahta hai
13.	Sentence: He wants to go to see the movie. Candidate: wah film dekhna jana chahta hai Reference: wah film dekhne jana chahta hai

Table 5: Twenty-eight randomly selected sentences

$S.No$	$Sentence, CandidateandReference$
14.	Sentence: That man wishes to buy a car. Candidate: vah adami ek car kharidna chahta hai Reference: vah adami ek car kharidna chahta hai
15.	Sentence: Has he decided to visit the museum? Candidate: Kya wah mueseum daura karna faisla kar chuka hai Reference: Kya wah mueseum ka daura karne ka faisla kar chuka hai
16.	Sentence: Where does he want to go to play? Candidate: wah kahaan khelna jana chahta hai Candidate: wah kahaan khelna jana chahta hai
.17	Sentence: Why does he not want to play? Candidate: wah kyon khelna nahi chahta hai Reference: wah kyon khelna nahi chahta hai
18.	Sentence: My friend wants to go. Candidate: mera dost jana chahta hai Reference: mera dost jana chahta hai
19.	Sentence: Why the old man did not tell us the truth? Candidate: boodha adami hamko sach kyon nahi bataya tha Reference: boodhe adami ne hamko sach kyon nahi bataya tha
20.	Sentence: Why did the old man buy a watch? Candidate: boodha adami ek ghadi kyon kharida tha Reference: boodhe adami ne ek ghadi kyon kharidi thi
21.	Sentence: The teacher did not give us homework. Candidate: ustad hamko ghar ke liye kam nahi diya tha Reference: ustad ne hamko ghar ke liye kam nahi diya tha
22.	Sentence: Shimla offers you refreshing environment. Candidate: SHIMLA tumko tazagi bhara mahaul deta hai Reference: Shimla tumko tazagi bhara mahaul deta hai
23.	Sentence: Have I given you my pen? Candidate: Kya mai tumko mera kalam de chuka hun Reference: Kya mai tumko apna kalam de chuka hun
24.	Sentence: The boy has lost his pen. Candidate: ladaka apni kalam kho chuka hai Reference: ladaka apni kalam kho chuka hai
25.	Sentence: What these boys were doing? Candidate: yen ladaken kya kar rahe the Reference: yen ladaken kya kar rahe the
26.	Sentence: Do the birds fly? Candidate: Kya chidiyan udati hain Reference: Kya chidiyan udati hain
27.	Sentence: The old man was working. Candidate: boodha adami kam kar raha tha Reference: boodha adami kam kar raha tha
28.	Sentence: Mr S Khan is a research scholar. Candidate: MR S KHAN ek taftish alam hai Reference: Mr S Khan ek taftish alam hai

this equation is calculated as Countclip = min (Count, Max Ref Count). The formula for calculating brevity penalty is

BP = 1 if c >r ; $BP = e^{(1-r/c)}$ if c $\leq$ r

Where r is the length of reference and c is the length of candidate; Then Bleu score is calculated as:

$BLUE = BP.exp(\Sigma w_n log p_n)$

Precision in [23] is the fraction of correct instances among those that the algorithm believes to belong to the relevant subset. Precision can be calculated as: P = | X ∩ Y | / | Y | Where Y is the set of candidate items and X is the of reference items.

Recall in [23] is the fraction of correct instances among all instances that actually belong to the relevant subset. Recall can be calculated as: R = | X ∩ Y | / | X | Where Y is the set of candidate items and X is the of reference items.

METEOR (Metric for Evaluation of Translation with Explicit ORdering) is a machine translation evaluation metric developed at Carnegie Mellon University. The Meteor metric is based on the weighted harmonic mean of unigram precision (P = m/w_t) and unigram recall (P = m/w_r). Where m is number of unigrams, w_t is the number of unigrams in candidate translation and w_r is the reference translation. Precision and Recall are combined using the harmonic mean with recall 9 times more than precision: F_{mean} = 10PR / 9P + R This measure is for congruity with respect to single words but for considering longer n-gram matches, a penalty p is calculated for the alignment as: p = 0.5 (c / $u_m)^3$; Where c is the number of chunks, and u_m is the number of unigrams that have been mapped. The more mappings there are that are not adjacent in the reference and the candidate sentence, the higher the penalty will be. Final Meteor-score (M-score) is calculated as:

M = F_{mean} (1-p).

F-Measure in [23] is a metric developed on the New York University. The F-measure is defined as the harmonic mean of precision and the recall as:

F-measure = (2 * Precision * Recall) / (Precision + Recall).

The comparative scores of different Machine Translation evaluation methods such as BLEU (BiLingual Evaluation Understudy), METEOR (M), F-measure (F) scores, unigram Precision (P), unigram Recall (R) for thirty-seven randomly selected sentences of various classes are shown in figure 4 (to see Figure4).

It has been seen from the results that system performs efficiently on those classes of sentences whose grammar rules are trained in the neural network. System uses Stanford Parser for typed dependency and Tagger for POS-tagging; if the parser or tagger makes an error for any sentence then same error will be propagated throughout the translation and will result in the wrong translation. We obtained an average BLUE score of 0.6954, M-score of 0.8583 and F-score of 0.8650.

6 Conclusion and Future Work

The working and architecture of our English to Urdu Machine translation system is discussed in this paper. All the modules have been implemented successfully. This paper describes the use of neural network with rule based machine translation approach. Our system uses neural network for dictionary lookups and grammar structure mapping and suffix addition to verbs and

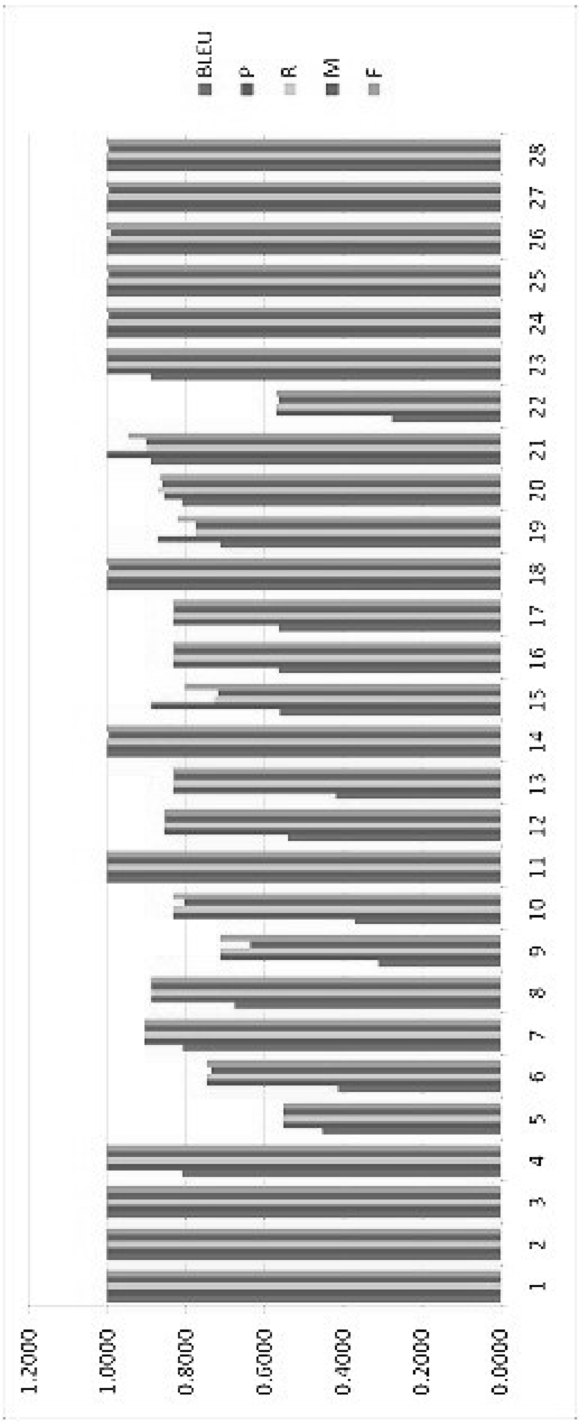


Figure 4: Evaluation Scores for randomly selected sentences

case marker addition with subject does not require any dictionary lookups, though it is done on the basis of information attached with the words; which makes it efficient and fast. Our system works efficiently on the sentences for which grammar rules are present in the rule base and words which are available in the bilingual dictionary. If the word is not present in the dictionary, English word is printed as it is in the translation in capitals. The translation results obtained from the system evaluated using machine evaluation methods and manually and it has seen that the system works efficiently on the trained linguistic rules and bilingual dictionary. The n-gram blue score obtained for the system over 100 sentences is 0.6954; METEOR score achieved is 0.8583 and F-score of 0.8650. So an enhancement to the grammar rules and size of bilingual dictionary will lead to the efficient and accurate machine translation system.

References

[1] Abdullah, P. and Homiedan, H. Machine translation, 1997.

[2] Ahmed, T. The interaction of light verbs and verb classes of urdu. In *Interdisciplinary workshop on Verbs: The identification and representation of verb features*, 2010.

[3] Atish Durrani, M. Q. Z. *Urdu informatics*. Islamabad : Center of Excellence for Urdu Informatics, National Language Authority, 2008.

[4] Hagan, M. and Menhaj, M. Training feedforward networks with the marquardt algorithm. *Neural Networks, IEEE Transactions on*, 5(6):989 –993, 1994.

[5] Hardie, A. Developing a tagset for automated part-of-speech tagging in urdu. In *Corpus Linguistics 2003*, 2003.

[6] Hock, H. *Principles of historical linguistics*. Mouton de Gruyter, 1986.

[7] Hutchins, W. *Machine Translation: past, present, future*. Chichester : Ellis Horwood, 1986.

[8] Imperial, N. K., Koncar, N., and Guthrie, D. G. A natural language translation neural network. In *In Proceedings of the International Conference on New Methods in Language Processing (NeMLaP*, pages 71–77, 1994.

[9] Jain, A. N. Parsing complex sentences with structured connectionist networks. *Neural Computation*, 3:110–120, 1991.

[10] Kashif Shaikh, M., Ali Khowaja, H., and Ahmed Khan, M. Urdu text translation with natural language processing. In *Engineering, Sciences and Technology, Student Conference On*, pages 81 – 85, 2004.

[11] Khan, S., Pervez, Z., Mahmood, M., Mustafa, F., and Hasan, U. An expert system driven approach to generating natural language in romanized urdu from english documents. In *Multi Topic Conference, 2003. INMIC 2003. 7th International*, pages 361 – 366, 2003.

[12] Mishra, V. and Mishra., R. Ann and rule based model for english to sanskrit machine translation. *INFOCOMP Journal of Computer Science*, 9(1):80–89, 2010.

[13] Muhammad, U., Bilal, K., Khan, A., and Khan, M. N. Aghaz: An expert system based approach for the translation of english to urdu. *International Journal of Social Sciences*, 3(1):70–74, 2008.

[14] Naim, C. and Qaumi Kaunsil bara'e Taraqqi-yi Urdu (New Delhi, I. *Introductory Urdu*. Number v. 1. National Council for Promotion of Urdu Language, 2000.

[15] Papineni, K., Papineni, K., Roukos, S., Roukos, S., Ward, T., Ward, T., jing Zhu, W., and jing Zhu, W. Bleu: A method for automatic evaluation of machine translation. pages 311–318, 2002.

[16] Parser, S. Stanford parser, http://nlp.stanford.edu/software/lex-parser.shtml, 2011.

[17] SIL. Sil international, 2011.

[18] Sinha, R. M. K. Developing english-urdu machine translation via hindi, 2009.

[19] Tafseer Ahmed, S. A. English to urdu translation system, 2002.

[20] Tagger, S. http://nlp.stanford.edu/software/tagger.shtml, 2011.

[21] Tahir, G., Asghar, S., and Masood, N. Knowledge based machine translation. In *Information and Emerging Technologies (ICIET), 2010 International Conference on*, pages 1 –5, 2010.

[22] TDIL. Technology development for indian languages programme, http://tdil-dc.in, 2011.

[23] Turian, J., Shen, L., and Melamed, I. D. Evaluation of machine translation and its evaluation. In *In Proceedings of MT Summit IX*, pages 386–393, 2003.

[24] Waibel, A., Jain, A., McNair, A., Saito, H., Hauptmann, A., and Tebelskis, J. Janus: a speech-to-speech translation system using connectionist and symbolic processing strategies. In *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, pages 793 –796 vol.2, 1991.

[25] Zafar, M. and Masood, A. Interactive english to urdu machine translation using example-based approach, 2009.