Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 485737, 8 pages
http://dx.doi.org/10.1155/2014/485737

Research Article

Quantum Neural Network Based Machine Translator for Hindi to English

Ravi Narayan,[1] V. P. Singh,[1] and S. Chakraverty[2]

[1] *Department of Computer Science & Engineering, Thapar University, Patiala, Punjab 147 004, India*
[2] *Department of Mathematics, National Institute of Technology Rourkela, Odisha 769 008, India*

Correspondence should be addressed to Ravi Narayan; ravi_n_sharma@hotmail.com

Received 31 August 2013; Accepted 8 January 2014; Published 27 February 2014

Academic Editors: P. Bala and J. Silc

This paper presents the machine learning based machine translation system for Hindi to English, which learns the semantically correct corpus. The quantum neural based pattern recognizer is used to recognize and learn the pattern of corpus, using the information of part of speech of individual word in the corpus, like a human. The system performs the machine translation using its knowledge gained during the learning by inputting the pair of sentences of Devnagri-Hindi and English. To analyze the effectiveness of the proposed approach, 2600 sentences have been evaluated during simulation and evaluation. The accuracy achieved on BLEU score is 0.7502, on NIST score is 6.5773, on ROUGE-L score is 0.9233, and on METEOR score is 0.5456, which is significantly higher in comparison with Google Translation and Bing Translation for Hindi to English Machine Translation.

1. Introduction

Machine translation is one of the major fields of NLP in which the researchers are having their interest from the time computers were invented. Many machine translation systems are available with their pros and cons for many languages. Researchers have also presented different approaches for computer to understand and generate the languages with semantics and syntactics. But still many languages are having translation difficulties due to ambiguity in their words and the grammatical complexity. The machine translator should address the key characteristic properties which are necessary to increase the performance of machine translation up to the level of human performance in translation. Most of the machine translators are working on the alignment of words in chunk (sentence).

This paper presents the quantum neural based machine translation for Hindi to English. The quantum neural network (QNN) based approach increases the accuracy during the knowledge adoptability. In this work our main focus is to show the significant increase in the accuracy of machine translation during our research with the pair of Hindi and English sentences. The machine translation is done using the new approach based on quantum neural network which learns the patterns of language using the pair of sentences of Hindi and English.

Some researchers have done their machine translation (MT) using statistical machine translation (SMT). The SMT uses the pattern recognition for automatic machine translation systems for available parallel corpora. Statistical machine translation needs alignment mapping of words between the source and target sentence. On one hand alignments are used to train the statistical models and, on the other hand, during the decoding process to link the words in the source sentence to the words of target sentence [1–4]. But SMT methods are having the problem of word ordering. To overcome the problem of word ordering and for increasing the accuracy, some researchers introduced the concept of syntax-based reordering for Chinese-to-English and Arabic-to-English [5].

Recently some work has been done with Hindi by several researchers using different methods of machine translation, like example based system [5, 6], rule based [7], statistical machine translation [8], and parallel machine translation system [9]. A. Chandola and Mahalanobis described the use of corpus pattern for alignment and reordering of words for English to Hindi machine translation using the neural network [10], but still there are a lot of possibilities to develop a MT System for Hindi to increase the accuracy of MT. Some of the important works on Hindi are discussed in Section 2.

The main motivation behind the study of QNN is the possibility to address the unrealistic situation as well as realistic situation, which is not possible with the traditional neural network. QNN learns and predicts more accurately and needs less computation power and time for learning in comparison to artificial neural network. Researchers introduced the novel approach of neural network model based on quanta states superposition, having multilevel transfer function [11–13].

The most important difference among classical neural network and QNN is of their respective activation functions. In QNN as a substitute of normal activation functions, a multilevel activation function is used. Each multilevel function consists of the summation of sigmoid functions excited with quantum difference [14].

In QNN, the multilevel sigmoid function has been employed as activation function and is expressed as

$$\operatorname{sgm}(x) = \frac{1}{n_s} \sum_{r=1}^{n_s} \left(\frac{1}{1 + \exp(x - \theta^r)} \right), \qquad (1)$$

where n_s denotes the total multilevel positions in the sigmoid functions and θ^r denotes quantum interval of quantum level r [2].

2. Hindi-English Machine Translation Systems

2.1. ANGLABHARTI-II Machine Translation System. ANGLABHARTI-II was proposed in 2004 which is a hybrid system based on the generalized example-base (GEB) with raw example-base (REB). At the time of development, the author establishes that the alteration in the rule-base is hard and the outcome is possibly random. This system consists of error-analysis component and statistical language component for postediting. Preediting component can change the entered sentence to a structure, to translate without difficulty [5].

2.2. MATRA Machine Translation System. The MaTra was introduced in 2004 which is based on transfer approach using a frame-like structured representation. In this the rule-based and heuristics approach is used to resolve ambiguities. The text classification module is used for deciding the category of news item before working in entered sentence. The system selects the appropriate dictionary based on domain of news. It requires human assistance in analyzing the input. This system also breaks up the complex English sentence to easy sentences, after examining the structure, it produces Hindi sentences. This system is developed to work in the domain of news, annual reports, and technical phrases [7].

2.3. Hinglish Machine Translation System. Hinglish machine translation system is developed in 2004 for standard Hindi to standard English. This system is developed to incorporate the added enhancement to available AnglaBharti-II MT System for English to Hindi and to AnuBharti-II systems for Hindi to English translation, developed by Sinha. The accuracy of this system is satisfactory more than 90%. As the verbs have multiple meanings, it is not able to determine the sense, due to nondeep grammatical analysis [6].

2.4. IBM-English-Hindi Machine Translation System. IBM-English-Hindi MT System is developed by IBM India Research Lab in 2006; at the beginning of this project they started to develop an example based MT system but later on shifted to the statistical machine translation system from English to Indian languages.

2.5. Google Translate. Google Translate was developed by Franz-Josef Och in 2007. This model used the statistical MT approach to translate English to other languages and vice versa. Among the 57 languages, Hindi and Urdu are the only Indian languages present with Google Translate. Accuracy of the system is good enough to understand the sentence after translation [15].

3. Proposed Machine Translation System for Hindi to English

The proposed machine translation (MT) system consists of two approaches, one is rule based MT system and the other is quantum neural based MT system. The source language goes into the rule based MT system and passes through the QNN based MT system to refine the MT done by rule based MT module, which basically recognizes and classifies the sentence category. 2600 sentences are used with English and their corresponding Devanagari-Hindi sentences. Each Devanagari-Hindi sentence consists of words with question word, noun, helping verb, negative word, verb, preposition, article, adjective, postnoun, adverb, and so forth. Each English sentence contains a question word, noun, helping verb, negative word, verb, preposition, article, adjective, postnoun, adverb, and so forth. The data used to train is produced by an algorithm, which is based on simple deterministic grammar. The entire architecture of the proposed MT system model is given in Figure 1.

4. Quantum Neural Architecture

As shown in the Figure 2, three-layer architecture of QNN consist of inputs, one layer of multilevel hidden units, and output layer. In QNN as a substitute of normal activation functions, a multilevel activation function is used. Each multilevel function consists of summation of sigmoid functions excited with quantum difference.

Where n_s denotes total multilevel positions in sigmoid functions, θ^r denotes quantum interval of quantum level r:

$$\operatorname{sgm}(x) = \frac{1}{n_s} \sum_{r=1}^{n_s} \left(\frac{1}{1 + \exp(-x \pm \theta^r)} \right). \qquad (2)$$

Here every neural network node represents three substates in itself with the difference of quantum interval θ^r with quantum

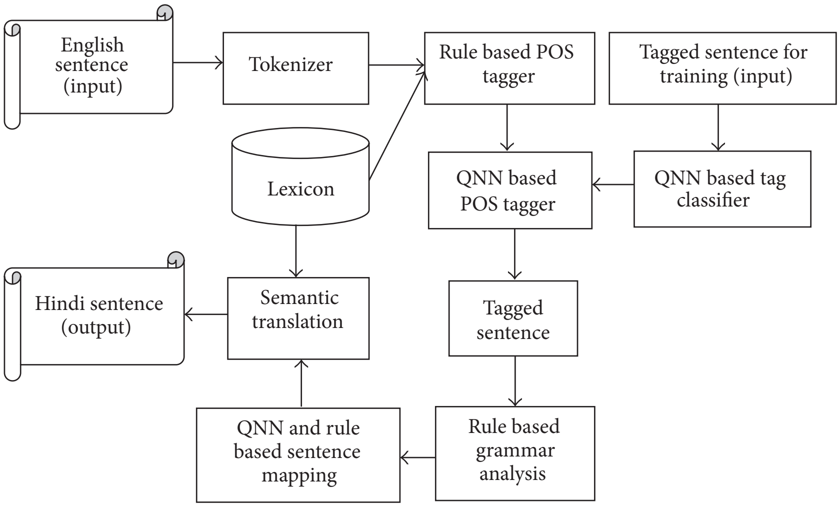

Figure 1: Architecture of MT system model.

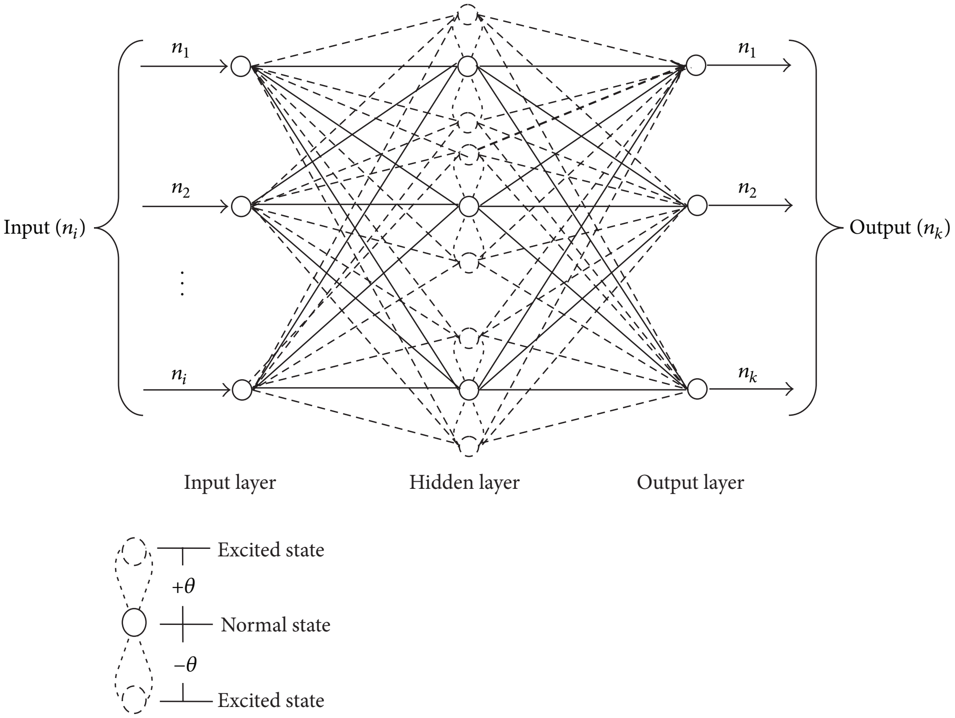

Figure 2: Architecture of Quantum Neural Network.

level r, where n_s denotes the number of grades in the quantum activation functions.

5. Quantum Neural Implementation of Translation Rules

The strategy is to first identify and tag the parts of speech using Table 1 and then translate the English (source language) sentences literally into Devanagari-Hindi (target language) with no rearrangement of words. After syntactic translation, rearrangement of the words has been done for accurate translation retaining the sense of translated sentence. The rules are based on parts of speech, not based on meaning. To facilitate the procedure, distinctive three-digits codes based on their parts of speech are assigned which are shown in Table 1.

For a special case when input sentence and the resulting sentence are having unequal number of words, then the dummy numeric code .000 is used for giving a similar word alignment.

Table 1: Numeric codes for parts of speech.

Parts of speech (subclass)	Numeric code
Prenoun (PN)	.100
Noun-infinitive (Ni)	.101
Pronoun (PRO)	.102
Gerund (GER)	.103
Relative pronoun (RPRO)	.104
Postnoun (POSTN)	.105
Verb (V)	.110
Helping verb (HV)	.111
Adverb (ADV)	.112
Auxiliary verb (AUX)	.113
Interrogative (question word) (INT)	.120
Demonstrative words (DEM)	.121
Quantifier (QUAN)	.122
Article (A)	.123
Adjective (ADJ)	.130
Adjective-particle (ADJP)	.131
Number (N)	.132
Preposition (PRE)	.140
Postposition (POST)	.141
Punctuation (PUNC)	.150
Conjunction (CONJ)	.160
Interjection (INTER)	.170
Negative word (NE)	.180
Determiner (D)	.190
Idiom (I)	.200
Phrases (P)	.210
Unknown words (UW)	.220

Case 1. When input sentence and the resulting sentence are having unequal numbers of words.

The coded version of sentence is thus

Ram	will	not	go	to	the	market.
.100	.111	.220	.110	.140	.123	.102

Therefore, input numeral sequence is [.100 .111 .220 .110 .140 .123 .102] and the corresponding output is [.100 .102 .220 .110 .111 .000 .000]. The outcome of neural network might not be the perfect integer, it should be round off and few basic error adjustments might be needed to find the output numeral codes. Even the network is likely to arrange the location of 3-digit codes. By this, it learns the target language knowledge which is needed for semantic rearrangement and also helps in parts of speech tagging, by pattern matching: it is also helpful to adopt and learn the grammar rules up to a level. For handling the complex sentences the algorithm is used. The algorithm first removes the interrogative and negative words, on the basis of conjunction; the system breaks up and converts the complex sentence into two or more small simple sentences. After the translation of each of the simple sentences, the system again rejoins the entire subsentences and also adds the removed interrogative and negative words in the sentence. The whole process is explained in Algorithm 1 in the next section.

5.1. Algorithm for Proposed QNN Based MT System for Complex Sentences

QNNMTS (SENTENCE, TOKEN, N, LOC). Here SENTENCE is an array with N elements containing Hindi words. Parameter TOKEN contains the token of each word and LOC keeps track of position. ICOUNT contains the maximum number of interrogative words encountered in sentence, NCOUNT contains the maximum number of negative words encountered in the sentence, and CCOUNT contains the maximum number of conjunction words encountered in the sentence. (see Algorithm 1).

6. Experiment and Results

All words in each language are assigned with a unique numeric code on the basis of their respective part of speech. Experiments show that memorization of the training data is occurring. The results shown in this section are achieved after training with 2600 Devanagari-Hindi sentences and their English translations. 500 tests are performed with the system for each value of quantum interval (θ) with random data sets selected from 2600 sentences; the dataset is divided in 4 : 3 : 3 ratios,respectively, for training, validation, and test from 2600 English sentences and their Devanagari-Hindi translations. In Table 2, the values are the average of 500 tests performed with the system for each value of quantum interval (θ) for 2600 sentences. The best performance is shown for value of quantum interval (θ) equal to one with respect to all the parameters; that is, epoch or iterations needed to train the network, the training performance, validation performance, and test performance in respect to their mean square error (MSE). Here it is clearly shown that QNN at (θ) equal to one is very much efficient as compared to classical artificial neural network at (θ) equal to zero. Table 2 clearly shows the comparison between the performances of QNN with ANN in respect to above said performance parameters and as a result we can conclude that QNN is better than ANN for machine translation.

7. Evaluations and Comparison

This paper proposed a new machine translation method which can combine the advantage of quantum neural network. 2600 sentences are used to analyze the effectiveness of the proposed MT system.

The performance of proposed system is comparatively analyzed with Google Translation (http://translate.google.com/) and Microsoft's Bing Translation (http://www.bing.com/translator) by using various MT evaluation methods like BLEU, NIST, ROUGE-L, and METEOR. For evaluation purpose we translate the same set of input sentences by using our proposed system, Google Translation, and Bing Translation and then evaluate the output got from each of

```
Step 1. Check whether the sentence is Complex sentence or Simple Sentence
        [Initialize.] Set ICOUNT = 1, NCOUNT = 1, CCOUNT = 1, ILOC, NLOC, CLOC.
        Repeat for LOC = 1 to $N$:
                if SENTENCE[LOC] = "Interrogative word", then:
                        Set ILOC[ICOUNT] := LOC,
                        ICOUNT = ICOUNT + 1.
                    [End of if structure]
                if SENTENCE[LOC] = "Negative word", then:
                        Set NLOC[NCOUNT] := LOC,
                        NCOUNT= NCOUNT + 1.
                    [End of if structure]
                if SENTENCE[LOC] = "Conjunction", then:
                        Set CLOC[CCOUNT] := LOC.
                        CCOUNT = CCOUNT + 1
                    [End of if structure]
        [End of for]
        if ILOC = NULL, or NLOC = NULL, or CLOC = NULL, then:
                Go to Step 5.
Step 2.         Remove the interrogative word from the complex sentence to make it Affirmative.
                Repeat for $X = 1$ to ICOUNT
                        Set ITemp[$X$]:= SENTENCE[ILOC[$X$]].
                        Set SENTENCE[ILOC[$X$]]:= Null.
                [End of for]
Step 3. Then Remove the negative to make it simple sentence
                Repeat for $Y = 1$ to NCOUNT
                        Set NTemp[$Y$]:= SENTENCE[NLOC[$Y$]].
                        Set SENTENCE[NLOC[$Y$]]:= Null.
                [End of for]
Step 4. Split the sentence into two or more simple sentences on the basis of conjunction.
                Repeat for $Z = 1$ to CCOUNT
                        Set CTemp[$Z$]:= SENTENCE[CLOC[$Z$]].
                        Set SENTENCE[CLOC[$Z$]]:= Hindi Full-stop ("|").
                [End of for]
Step 5.     Pass each sub sentence with TOKEN to QNN based Machine Translator for reposition.
Step 6. Refine the Translated sentences by applying the grammar rules.
Step 7. Add the interrogative word if removed in Step 2.
                Repeat for $X = 1$ to ICOUNT
                        if ITEMP[$X$] = NOT NULL
                Set SENTENCE[ILOC[$X$]]:= ITemp[$X$].
                        [End of if structure]
                [End of for]
Step 8. Add the negative word if removed in Step 3.
                Repeat for $Y = 1$ to NCOUNT
                        if NTEMP[$Y$] = NOT NULL
              Set SENTENCE[NLOC[$Y$]]:= NTemp[$Y$].
                        [End of if structure]
                [End of for]
Step 9. Rejoin the entire sub sentences, if split in Step 4.
Step 10. Semantic Translation
Step 11.    Exit.
```

Algorithm 1

the systems. The fluency check is done by *n*-gram analysis using the reference translations.

7.1. BLEU. We have used BLEU (bilingual evaluation understudy) to calculate the score of system output. BLEU is an IBM developed metric, which uses modified n-gram precision to compare the candidate translation against reference translations [16].

Comparative bar diagram between proposed system, Google, and Bing based on BLEU scale is shown in Figure 3. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.7502 on BLEU scale, Bing

Table 2: Comparison of performance measurement of MT system with QNN and traditional neural network.

S. No	Quantum interval (θ)	Epoch (iteration)	Training performance (MSE)	Validation performance (MSE)	Test performance (MSE)
1	$\theta = 0$ is equivalent to ANN	20.04391	0.002027	0.002066	0.002075
2	0.25	15.36727	0.000185	0.000205	0.000203
3	0.5	15.47305	0.000248	0.000273	0.000273
4	0.75	15.28343	0.000231	0.000248	0.000252
5	**1***	**15.48303**	**0.00025**	**0.00027**	**0.000272**
6	1.25	15.67066	0.000284	0.0003	0.0003
7	1.5	15.64271	0.000296	0.000318	0.000316
8	1.75	15.78643	0.000349	0.000374	0.000375
9	2	15.68663	0.000184	0.000204	0.000209
10	2.25	15.96607	0.000249	0.000273	0.000273
11	2.5	16.27345	0.000256	0.000281	0.000276
12	2.75	16.54092	0.00022	0.000242	0.000242
13	3	16.49301	0.000348	0.000361	0.00037
14	3.25	16.93214	0.000192	0.000214	0.000216
15	3.5	17.76248	0.000294	0.000315	0.000323
16	3.75	17.85429	0.000185	0.000202	0.000207
17	4	18.72056	0.000217	0.000237	0.00024

*For the value of $\theta = 1$, the optimum trade-off of iteration and the minimum Error (MSE) achieved.

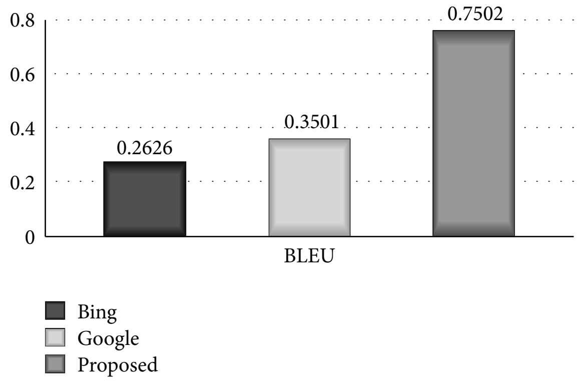

Figure 3: Comparative bar diagram between proposed system, Google, and Bing based on BLEU scale.

has shown accuracy of 0.2626, and Google has shown 0.3501 accuracy on BLEU scale.

Calculate the n-gram resemblance by comparing the sentences. Then add the clipped n-gram counts for all the candidate sentences and divide by the number of candidate n-grams in the test sentence to calculate the precision score, p_n, for the whole test sentence

$$p_n = \frac{\sum_{C\in\{\text{Candidates}\}} \sum_{n\text{-gram}\in C} \text{Count}_{\text{clip}}(n\text{-gram})}{\sum_{C'\in\{\text{Candidates}\}} \sum_{n\text{-gram}'\in C'} \text{Count}_{\text{clip}}(n\text{-gram}')}, \quad (3)$$

where $\text{Count}_{\text{clip}} = \min(\text{Count}; \text{Max Ref Count})$. In other words, one truncates each word's count.

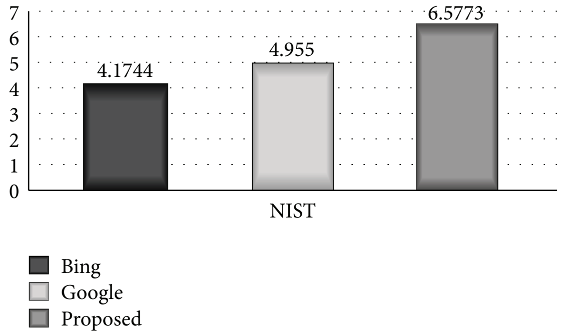

Figure 4: Comparative bar diagram between proposed system, Google, and Bing based on NIST scale.

Here c denotes length of the candidate translation and r denotes reference sentence length. Then calculate brevity penalty BP:

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \le r. \end{cases} \quad (4)$$

Then,

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right). \quad (5)$$

7.2. NIST. Proposed by NIST (national institute of standard and technology), it reduces the effect of longer N-grams by using arithmetic mean over N-grams counts instead of

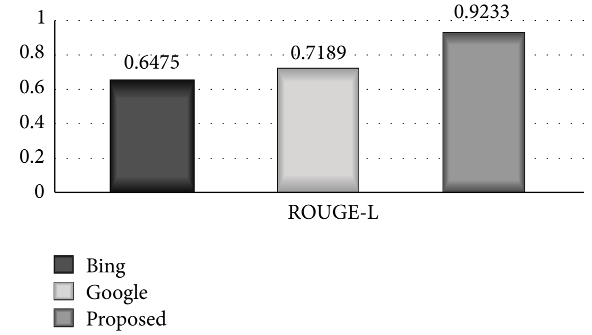

Figure 5: Comparative bar diagram among proposed system, Google, and Bing based on ROUGE-L scale.

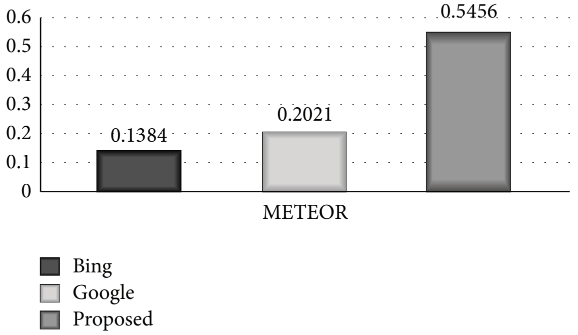

Figure 6: Comparative bar diagram among proposed system, Google, and Bing based on METEOR scale.

geometric mean of cooccurrences over N [17]. Figure 4 shows the comparative bar diagram between proposed system, Google, and Bing based on NIST scale. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 6.5773 on NIST scale, Bing has shown accuracy of 4.1744, and Google has shown 4.955 accuracy on NIST scale

$$\mathrm{NIST}_{\mathrm{score}} = \mathrm{BP}_{\mathrm{NIST}} * \mathrm{PRECISION}_{\mathrm{NIST}},$$

$$\mathrm{PRECISION}_{\mathrm{NIST}} = \sum_{n=1}^{N} \left\{ \frac{\sum_{\mathrm{all\_}w_1\cdots w_n\mathrm{\_that\_co\text{-}occur}} \mathrm{Info}\,(w_1 \cdots w_n)}{\sum_{\mathrm{all\_}w_1\cdots w_n\mathrm{\_in\_hypo}} (1)} \right\}, \tag{6}$$

where

$$\mathrm{Info}\,(w_1 \cdots w_n) = -\log_2 \left(\frac{\mathrm{Count}\,(w_1 \cdots w_n)}{\mathrm{Count}\,(w_1 \cdots w_{n-1})} \right), \tag{7}$$

where Info weights more the words that are difficult to predict and count is computed over the full set of references; theoretically the precision range is having no limit

$$\mathrm{BP}_{\mathrm{NIST}} = \exp \left\{ \beta * \log^2 \left[mn \left(\frac{\mathrm{LenHypo}}{\mathrm{LenRef}}, 1 \right) \right] \right\}, \tag{8}$$

where LenHypo is total length of hypothesis and LenRef is average length of all references which does not depend on hypothesis.

7.3. ROUGE-L. ROUGE-L (recall-oriented understudy for gisting evaluation-longest common subsequence) calculates the sentence-to-sentence resemblance using the longest common substring among the candidate translation and reference translations. The longest common substring represents the similarity among two translations. F_{lcs} calculates the resemblance between two translations X of length m and Y of length n; X denotes reference translation and Y denotes candidate translation [18]. Comparative bar diagram between proposed system, Google, and Bing based on ROUGE-L scale is shown in Figure 5. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.9233 on ROUGE-L scale, Bing has shown accuracy of 0.6475, and Google has shown 0.7189 accuracy on ROUGE-L scale

$$R_{\mathrm{lcs}} = \frac{\mathrm{LCS}\,(X, Y)}{m},$$
$$P_{\mathrm{lcs}} = \frac{\mathrm{LCS}\,(X, Y)}{n}, \tag{9}$$
$$F_{\mathrm{lcs}} = \frac{\left(1 + \beta^2\right) R_{\mathrm{lcs}} P_{\mathrm{lcs}}}{R_{\mathrm{lcs}} + \beta^2 P_{\mathrm{lcs}}},$$

where P_{lcs} is precision and R_{lcs} is recall and $\mathrm{LCS}(X, Y)$ denotes the longest common substring of X and Y, and $\beta = P_{\mathrm{lcs}}/R_{\mathrm{lcs}}$ when $\partial F_{\mathrm{lcs}}/\partial R_{\mathrm{lcs}} = \partial F_{\mathrm{lcs}}/\partial P_{\mathrm{lcs}}$

$$\text{Rouge-L} = \text{Harmonic Mean}\,(P_{\mathrm{lcs}}, R_{\mathrm{lcs}}) = \frac{(2 * P_{\mathrm{lcs}} * R_{\mathrm{lcs}})}{(P_{\mathrm{lcs}} + R_{\mathrm{lcs}})}. \tag{10}$$

7.4. METEOR. METEOR (metric for evaluation of translation with explicit ordering) is developed at Carnegie Mellon University. Figure 6 shows comparative bar diagram between proposed system, Google, and Bing based on METEOR scale. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.5456 on METEOR scale, Bing has shown accuracy of 0.1384, and Google has shown 0.2021 accuracy on METEOR scale.

The METEOR weighted harmonic mean of unigram precision ($P = m/w_t$) and unigram recall ($R = m/w_r$) used.

Here m denotes unigram matches, w_t denotes unigrams in candidate translation, and w_r is the reference translation. F_{mean} is calculated by combining the recall and precision via a harmonic mean that places equal weight on precision and recall as ($F_{\mathrm{mean}} = 2PR/(P + R)$).

This measure is for congruity with respect to single word but for considering longer n-gram matches; a penalty p is calculated for the alignment as ($p = 0.5(c/u_m)^3$).

Here c denotes the number of chunks and u_m denotes the number of unigrams that have been mapped [19].

Final METEOR-score (M-score) can be calculated as follows:

$$\text{Meteor-score} = F_{\text{mean}} \left(1 - p\right). \tag{11}$$

Experiments confirm that the accuracy was achieved for machine translation based on quantum neural network, which is better than other bilingual translation methods.

8. Conclusion

In this work we have presented the quantum neural network approach for the problem of machine translation. It has demonstrated the reasonable accuracy on various scores. It may be noted that BLEU score achieved 0.7502, NIST score achieved 6.5773, ROUGE-L score achieved 0.9233, and METEOR score achieved 0.5456 accuracy. The accuracy of the proposed system is significantly higher in comparison with Google Translation, Bing Translation, and other existing approaches for Hindi to English machine translation. Accuracy of this system has been improved significantly by incorporating techniques for handling the unknown words using QNN. It is also shown above that it requires less training time than the neural network based MT systems.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Rodríguez, I. García-Varea, and A. José Gámez, "On the application of different evolutionary algorithms to the alignment problem in statistical machine translation," *Neurocomputing*, vol. 71, pp. 755–765, 2008.

[2] S. Ananthakrishnan, R. Prasad, D. Stallard, and P. Natarajan, "Batch-mode semi-supervised active learning for statistical machine translation," *Computer Speech & Language*, vol. 27, pp. 397–406, 2013.

[3] D. Ortiz-Martínezb, I. García-Vareaa, and F. Casacubertab, "The scaling problem in the pattern recognition approach to machine translation," *Pattern Recognition Letters*, vol. 29, pp. 1145–1153, 2008.

[4] J. Andrés-Ferrer, D. Ortiz-Martínez, I. García-Varea, and F. Casacuberta, "On the use of different loss functions in statistical pattern recognition applied to machine translation," *Pattern Recognition Letters*, vol. 29, no. 8, pp. 1072–1081, 2008.

[5] M. Khalilov and J. A. R. Fonollosa, "Syntax-based reordering for statistical machine translation," *Computer Speech & Language*, vol. 25, no. 4, pp. 761–788, 2011.

[6] R. M. K. Sinha, "An engineering perspective of machine translation: anglabharti-II and anubharti-II architectures," in *Proceedings of the International Symposium on Machine Translation, NLP and Translation Support System (ISTRANS '04)*, pp. 134–138, Tata McGraw-Hill, New Delhi, India, 2004.

[7] R. M. K. Sinha and A. Thakur, "Machine translation of bi-lingual Hindi-English (Hinglish)," in *Proceedings of the 10th Machine Translation Summit*, pp. 149–156, Phuket, Thailand, 2005.

[8] R. Ananthakrishnan, M. Kavitha, J. H. Jayprasad et al., "MaTra: a practical approach to fully-automatic indicative English-Hindi machine translation," in *Proceedings of the Symposium on Modeling and Shallow Parsing of Indian Languages (MSPIL '06)*, 2006.

[9] S. Raman and N. R. K. Reddy, "A transputer-based parallel machine translation system for Indian languages," *Microprocessors and Microsystems*, vol. 20, no. 6, pp. 373–383, 1997.

[10] A. Chandola and A. Mahalanobis, "Ordered rules for full sentence translation: a neural network realization and a case study for Hindi and English," *Pattern Recognition*, vol. 27, no. 4, pp. 515–521, 1994.

[11] N. B. Karayiannis and G. Purushothaman, "Fuzzy pattern classification using feed-forward neural networks with multilevel hidden neurons," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1577–1582, Orlando, Fla, USA, June 1994.

[12] G. Purushothaman and N. B. Karayiannis, "Quantum Neural Networks (QNN's): inherently fuzzy feedforward neural networks," *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 679–693, 1997.

[13] R. Kretzschmar, R. Bueler, N. B. Karayiannis, and F. Eggimann, "Quantum neural networks versus conventional feedforward neural networks: an experimental study," in *Proceedings of the 10th IEEE Workshop on Neural Netwoks for Signal Processing (NNSP '00)*, pp. 328–337, December 2000.

[14] Z. Daqi and W. Rushi, "A multi-layer quantum neural networks recognition system for handwritten digital recognition," in *Proceedings of the 3rd International Conference on Natural Computation (ICNC '07)*, pp. 718–722, Hainan, China, August 2007.

[15] T. Brants, A. C. Popat, P. Xu, F. J. Och, and J. Dean, "Large language models in machine translation," in *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 858–867, Prague, Czech Republic, June 2007.

[16] K. Papineni, S. Roukos, T. Ward, and W. -J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL '02)*, pp. 311–318, Philadelphia, Pa, USA, 2002.

[17] G. Doddington, "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics," in *Proceedings of the 2nd International Conference on Human Language Technology Research*, 2002.

[18] C. Y. Lin, "Rouge: a package for automatic evaluation of summaries," in *Proceedings of the Workshop on Text Summarization Branches Out*, Barcelona, Spain, 2004.

[19] S. Banerjee and A. Lavie, "METEOR: an automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for MT*, Association of Computational Linguistics (ACL), Ann Arbor, Mich, USA, 2005.