Indonesian Journal of Electrical Engineering and Computer Science
Vol. 37, No. 1, January 2025, pp. 250~256
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v37.i1.pp250-256

Attention based English to Indo-Aryan and Dravidian language translation using sparsely factored NMT

Ritesh Kumar Dwivedi[1], Parma Nand[1], Om Pal[2]
[1]Department of Computer Science and Engineering, Sharda University, Greater Noida, India
[2]Department of Computer Science, University of Delhi, New Delhi, India

Article Info

Article history:

Received Mar 13, 2024
Revised Jun 11, 2024
Accepted Sep 2, 2024

Keywords:

BLEU scores
Linguistic dropout
Machine translation
NMT
SFNMT
Transformer

ABSTRACT

Neural machine translation (NMT) is a sophisticated technique that employs a large, singular neural network to learn and execute automatic translation tasks. Unlike statistical machine translation systems, NMT handles the entire translation process in an end-to-end manner, removing the need for additional components. This approach has shown significant promise in translation quality and has become the prevalent method. In this study, we apply sparsely factored NMT to English and several Indo-Aryan (Hindi, Bengali) and Dravidian (Tamil, Malayalam) language pairs. Specifically, we develop the machine translation system using an attention-based mechanism. A significant problem with traditional transformers is the huge memory requirement. Therefore, a sparsely factored NMT (SFNMT) is used to reduce the memory requirement but also improves the training time, thereby, reducing the computing time. In this paper, take inspiration from Vaswani transformer and modify it to get the best results. The system's performance was evaluated using the BLEU metric. The proposed model indtrl achieves a BLUE score of 32.13 (en→hi), 29.31 (en→be), 31.21 (en→ta), 21.12 (en→ml) and 32.67 (en→hi), 29.38 (en→be), 31.75 (en→ta), 21.17 (en→ml) without backtranslation and with backtranslation. To evaluate the performance of the system, we compared the results with those of existing systems. The developed system demonstrated a marginally higher BLEU score than both AnglaMT and Google translate.

Corresponding Author:

Ritesh Kumar Dwivedi
Department of Computer Science and Engineering, Sharda University
Greater Noida, India
Email: ritesh.dwivedi@nic.in

1. INTRODUCTION

"Machine translation" refers to the automated translation of text between languages using computers. In the field of artificial intelligence (AI), machine translation (MT) is considered an AI-complete problem, meaning that solving MT is akin to addressing the core challenge of AI: developing a system with general intelligence. Weaver (1949) presented the first workable method for translating text using computers. It stimulated the field of MT research. The original machine translation models relied solely on word-for-word replacements with multilingual dictionaries, which never produced satisfactory translations. The poor appraisal of the infamous 1966 ALPAC report1 hindered research on MT. The adoption of statistic-based methodologies marked a significant resurgence of research. These methods used a sentence-aligned parallel corpus to learn the bilingual dictionaries (or translation models) in a probabilistic manner. Statistical machine translation (SMT) models is the aggregate name for these models. For over a decade, SMT was the prevailing standard. Phrase-based models [1], while long-established and utilized in both commercial applications and machine translation research, have seen their translation quality plateau over time. The long-term dependencies in a sentence are not captured by phrase-based SMT models, which base their translation judgments on phrases. The addition of numerous components, including language models, reordering models, length penalties, and translation models, has made the entire SMT process more complex [2]. Owing to these challenges, a significant modification of the current system was necessary. Several of SMT systems' drawbacks are addressed by NMT. It is a comprehensive end-to-end system that uses large neural network to simulate the whole machine translation process. The researcher proposed a synchronous inference method capable of generating translations in multiple languages simultaneously [3]. current machine learning translation models, aside from "hi-en" translation, appear to translate other Indian languages (Bengali, Tamil, Punjabi, Urdu and Guajarati, Telugu, Kannada, Malayalam) with an accuracy of only 10%, Khan *et al.* [4]. A translation model from English to Tamil with a BLEU score of 8.33 has been proposed by Choudhary *et al*. [5].

An NMT system of IIIT-H has been proposed by Vikrant in a paper for the evaluation of WMT19. They translated news from Gujarati to English using an attention model for their work, and they received a BLEU score of 9.8. They have tackled the limited resource data as an issue. Good efforts have been made in translating English to Punjabi. Kamal Deep have proposed a model that claims to give a BELU score of 38.30 for Punjabi to English and 36.96 for English to Punjabi. The Punjabi language does not have as low of a resource corpus as it is for Bhojpuri or other languages. In another work by Singh *et al.* [6] Strong attention models were employed, and the results showed a BLEU score of roughly 24.48 for both types of translation. A work by Haque and Hasan [7] has proposed a model claiming that the sentences produced in translating English sentences to Bengali by their model are more semantically correct than the sentences produced by Google translator. Very few researches have been done in the English to Bengali language, and many of them are not up to par [7]. This research proposes some special translating rules because Bengali is a very hard language and producing right sentences is the true challenge, even though their algorithm has produced accuracy rates of above 97%. The paper by Sipra [8] findings on the subject of word borrowing from English to Urdu translation demonstrate that there are three possible approaches: direct borrowing with minimal or no modification, using a translator to translate from English to Urdu, and combining Urdu and English. He provides no detailed information on any NMT technique or methodology related to embedding vector designing in his work. Lingam *et al.* [9] proposed a rule-based method for translating Telugu into English. This approach, highlighted during a comprehensive review of studies in the field, demonstrated 92% translation accuracy for most sentences, while other sentences achieved around 50% accuracy. The construction of a more exotic model that is based on NMT principles and can generate sentences that are more semantically correct is where the gap is found in this instance.

In this paper, a new approach using transformer-based machine translation is shown. A significant problem with traditional transformers is the huge memory requirement. Therefore, a sparsely factored neural machine translation (SFNMT) is used that reduces the memory required and improves the training time, thereby, reducing the computing time. In this paper, take inspiration from Vaswani transformer and modify it to get the best results. The proposed model was trained on datasets of Samanantar. BLUE score helps in evaluation of the performance. The proposed model indtrl achieves a BLUE score of 32.13 (en→hi), 29.31 (en→be), 31.21 (en→ta), 21.12 (en→ml) and 32.67 (en→hi), 29.38 (en→be), 31.75 (en→ta), 21.17 (en→ml) without backtranslation and with backtranslation. A comparison with other models such as AnglaMT and Google translate was done and it was observed that the proposed model performs slightly (as per BLEU score) than the other models.

The structure of the paper is as follows: section 2 describes the methodology, beginning with an explanation of how SFNMT is employed, followed by details on the modified transformer. Next, the dataset details are presented. In section 3 outlines the experimental setup and results. Finally, section 4 discusses the conclusion and suggests future directions.

2. METHOD

2.1. Sparsely factored NMT

In our suggested method, unlike other systems that only receive plain text for translation, this model receives text annotated by a linguistic annotation system for source-side translation [10]-[12]. Raw text is used in the target side for training. The morphological features component has a wide range of possible values since each word might contain a mixture of these feature values. Its individual feature values may be rare, leading to infrequent updates to the embedded vectors for morphological characteristics during training. When splitting the data for training, the model shows the frequency of combinations of morphological features compared to the frequency of each individual morphological feature. It is evident that there are orders of magnitude more possible combinations than there are individual features.

It is suggested labelling each word according to the morphological feature space rather than treating each combination as a separate factor value. To do this, the model maintains an embedding table with a value for each morphological feature for each entry. In addition to the previously mentioned morphological feature vocabulary, a lemma-vocabulary is also maintained. Specifically, while encoding text for sparsely factored NMT, it is verified that every word is lemmatizable with inclusion in the vocabulary (lemma). If so, the model encodes the word by adding the lemma's embedded vector along with the embedded vectors of each morphological characteristic it possessed. In the event that a word cannot be lemmatized or is not found in the vocab, BPE is used for tokenization, an embedding table is also maintained. As a result, our tokens can be sub words or (lemma + morphological characteristics). The text is fed into a typical transformer model after it has been encoded as a series of embedded vectors.

Building upon the previously published base version, it is suggested an additional extension: a new hyperparameter called 'linguistic dropout' (LD) is introduced, representing the probability of tokenizing a word using a subword instead of the (lemma + morphological characteristics) representation. Both the lemmatized representation (if available) and the subword representation are created during data preparation. A Bernoulli distribution associated with the LD probability is used to determine the type of representation for each word during batch creation. The goal of LD is to train the model to operate in scenarios without linguistic data, such as when a word is not in its lexicon. Training with LD results in more frequent updates to the subword token embeddings, leading to more robust systems, especially when handling out-of-domain material.

2.2. Transformer model

Transformers have gained significant importance in translations or deeper neural networks shows a good performance [13]-[15] inspiration is taken from [16] and make modifications to the transformer. The encoder receives the following sequence $x = (x_1, x_2, x_3, \ldots, x_n)$ and maps it to $z = (z_1, z_2, z_3, \ldots, z_n)$. Given z, output is generated with the help of the decoder, the output is $y = (y_1, y_2, y_3, \ldots, y_n)$. Being auto-regressive at each step [17], using the symbols that were generated earlier as extra input for the following generation. Figure 1 of the transformer, respectively, details the transformer architecture with fully connected layers and layered self-attention, which adheres to this overall design. In the following sections, changes made to the [16] transformer are described. These changes were finalized as they proved the best efficiency of the model.

2.2.1. Stacks-encoder and decoder

- Encoder: to enhance capacity, a stack of N=8 layers is employed for the self-attention mechanism and positional encoding. Moreover, a change is made to the output dimension d_{model} from 512 to 256 to ensure faster training of the model. There are no other changes in the architecture.
- Decoder: here, a stack of N=8 identical layers is used. There are no other changes in the architecture.

2.2.2. Attention through scaled dot-product

The dot-product is given by the (1):

$$Attention(Q, K, V) = SoftMax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

2.2.3. Optimizer

Adam optimizer is used as it is originally in [16], however, the best results are with $\beta_1 = 0.92$ and $\beta_2 = 0.96$.

2.3. Dataset

Samanantar dataset, developed by AI4Bharat, is a multilingual parallel corpus designed to enhance machine translation systems for Indian languages. It consists of high-quality, aligned translations in multiple language pairs, including major Indian languages. The dataset was created to address the need for robust training data in low-resource language settings and aims to support the development of advanced machine translation models by providing extensive and diverse linguistic data. The Samanantar dataset is a significant resource for improving translation accuracy and performance in Indian languages, facilitating research and applications in natural language processing. In Table 1, four language pairs are shown with the no. of sentences for each language pair.

Parallel corpora shared by [18], [19] is leveraged and data augmented to enhance the translation quality for low resourced languages. Tokenizing sentences across all languages is done using trained models, and then filter sentences based on the ratio of source to target token count [20]. Researchers examine translation from very low-resource languages, such as Dzongkha to English, using various NMT models. The results indicate that the BLEU score of Seq2Seq models fluctuates, while the BLEU score of the transformer model shows a steady increase [21]. Comparing some system's accuracy to Google translate's 38.67% for the identical test samples, the researchers' results show that their approach is much more effective at 79.33% [22]. Beseiso *et al.* [23] introduce an innovative linguistic-based evaluation method for English-translated Arabic sentences, which significantly outperforms traditional MT evaluation methods like BLEU. As a pre-processing step for machine translation, Nyein and Soe [24] employed a model to reorder English phrases to match the word order of Myanmar, resulting in quality improvements on par with the baseline rule-based reordering technique. Ayu *et al.* [25] used in-depth linguistic knowledge ways to explain why a specific sequence knowledge/value parameter in the translation process performs better than the other method.

Table 1. Samananantar dataset description for Hindi, Oriya, Punjabi, and Tamil

Language pair	Parallel Corpus (No. of sentences)
English-Hindi	10125706
English- Bengali	8604580
English- Malayalam	5924426
English-Tamil	5264867

3. EXPERIMENT AND RESULTS

The model is trained on one machine with 8 NVIDIA A100 GPUs. In Table 2 describes the infrastructure details. To evaluate the importance of different components of the transformer, changes to the hyper-parameters of the base model were made in different ways, measuring the change in performance on the translations on the development set.

Table 3 shows BLEU scores and provides details of hyper parameters taken during model training. Following Figure 1 shows the comparison between proposed model and other standard models. It is showing better BLUE score for all four language pairs. Proposed model using SFNMT. It not only reduces the memory required but also improves the training time, thereby, reducing the computing time.

Table 4 compares the performance of five machine translation models for translating English to four languages. The results are measured using two metrics: BLEU score (indicating overall similarity to reference translations) and RIBES score (indicating fluency based on human judgment). English to Hindi (en-hi): Google translate leads with a BLEU score of 32.33, suggesting its translations are most similar to human references for this language pair. However, Indtrl achieves a strong score of 32.13, indicating very competitive performance. Interestingly, adding backtranslation to Indtrl improves its BLEU score slightly to 32.67. In terms of RIBES score, Indtrl (with or without backtranslation) outperforms all other models (0.7868 and 0.7872, respectively), suggesting its translations are judged as more fluent by humans for English-Hindi. English to Bengali (en-be): Indtrl takes the lead with a BLEU score of 29.31, followed by Google translate at 28.26. Backtranslation doesn't seem to significantly impact Indtrl's performance for Bengali (BLEU score remains at 29.38). When it comes to RIBES scores, Indtrl again dominates (0.7635) followed by Bing translation (0.7882). This indicates that Bing translation achieves a higher level of fluency for Bengali despite having a lower BLEU score. English to Tamil (en-ta): Google translate achieves a BLEU score of 30.98. Indtrl follows closely behind at 31.21, and backtranslation offers a minor improvement (31.75). In Table 5, a comparison of performance of indtrl was done with other standard models including Google translation, Bing translation, AI4Bharat IndicTrans translation.

Table 2. Experimental setup details

Resource type	Details
Processor	Intel Xeon
Random access memory	512 GB
Graphics processing unit	8x NVIDIA A100 GPUs
Language	Python

Table 3. BLUE scores with different configurations of the proposed transformer

	N	d_{model}	d_{ff}	h	d_k	d_v	P_{drop}	BLUE (en-hi)	BLUE (en-be)	BLUE (en-ta)	BLUE (en-ml)
Final	8	256	2048	8	64	64	0.1	32.13	29.31	31.21	21.12
				4	128	128		26.65	26.63	28.75	18.8
				16	32	32		26.85	28.29	27.78	19.53
	6							27.72	28.34	27.05	19.46
	4							26.22	28.19	28.25	19.28
	2							27.71	27.6	26.61	19.77

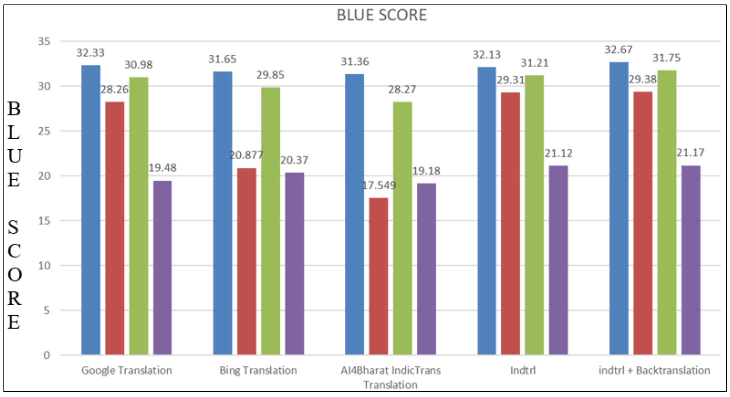

Figure 1. BLUE score of proposed model and other standard models

Table 4. Performance comparison of indtrl with other standard models

Model	BLUE (en-hj)	BLUE (en-be)	BLUE (en-ta)	BLUE (en-ml)
Google translation	32.33	28.26	30.98	19.48
Bing translation	31.65	20.877	29.85	20.37
AI4Bharat IndicTrans translation	31.36	17.549	28.27	19.18
Indtrl	32.13	29.31	31.21	21.12
Indtrl + Backtranslation	32.67	29.38	31.75	21.17

Table 5. Performance comparison of indtrl with other standard models

Model	RIBES (en-hj)	RIBES (en-be)	RIBES (en-ta)	RIBES (en-ml)
Google translation	0.6693	0.6175	0.7118	0.6334
Bing translation	0.6652	0.7882	0.6881	0.7307
AI4Bharat IndicTrans translation	0.6576	0.6221	0.6358	0.7803
Indtrl	0.7868	0.7635	0.7472	0.7253
Indtrl + Backtranslation	0.7872	0.7921	0.7491	0.7310

However, in Table 5, the RIBES score provides more insights. Indtrl (with and without backtranslation) achieves the highest scores (0.7472 and 0.7491), suggesting its translations are judged as more fluent for English-Tamil. English to Malayalam (en-ml): this is the most challenging translation for all models, reflected in the lower BLEU scores. Google translate leads with 19.48, followed by Indtrl at 21.12. Backtranslation offers minimal improvement for indtrl (21.17). Interestingly, AI4Bharat IndicTrans Translation achieves the highest RIBES score (0.7803) despite having a lower BLEU score. This suggests its translations is judged as more fluent despite being less similar to reference translations in terms of content. Overall, Indtrl with backtranslation emerges as a strong contender, particularly for achieving human-quality fluency as measured by RIBES. However, Google translate remains competitive for English-Hindi and Tamil based on BLEU score. For Malayalam translation, AI4Bharat IndicTrans seems to prioritize fluency based on RIBES score.

4. CONCLUSION

An attempt to translate English language into Hindi, Bengali, Tamil, and Malayalam was made with an innovative approach of using SFNMT and a modified transformer model. Moreover, to improve the dataset and translation, a method of backtranslation was applied. Changes to the transformer include the number of layers in encoder-decoder, changes in the output dimension, and changes in the hyperparameters of the Adam optimizer. The model named indtrl was compared with other standard models and performed slightly better and best with backtranslation. While indtrl with backtranslation emerges as a strong all-around performer, achieving good scores in both similarity (BLEU) and fluency (RIBES), other models have their strengths. Google translate excels in content similarity for English to Hindi and Tamil (high BLEU scores). However, BLEU and RIBES scores conflict each other. For instance, a model with high BLEU might not be the most fluent based on RIBES score. English to Malayalam translation seems most challenging, with AI4Bharat IndicTrans prioritizing fluency (high RIBES) even if content similarity is lower (lower BLEU).

Finally, backtranslation offers some improvement for indtrl but the impact is modes. A significant problem with traditional transformers is the huge memory requirement. Therefore, a SFNMT was used that not only reduced the memory required but also improved the training time, thereby, reducing the computing time. The proposed model was trained on datasets: Backtranslated-Hindi, IITB-hi-en, WAT-ILMPC, ILCI. The model achieves an average speed of 11.37 milliseconds per sentence, which otherwise was 14.56 milliseconds without the SFNMT. The proposed model indtrl achieves a BLUE score of 32.13 (en→hi), 29.31 (en→be), 31.21 (en→ta), 21.12 (en→ml) and 32.67 (en→hi), 29.38 (en→be), 31.75 (en→ta), 21.17 (en→ml) without backtranslation and with backtranslation. A limited dataset is a prime source of low BLUE scores; however, different configurations of the transformer model can provide better results. It is recommended to employ techniques such as grid search or random search to systematically explore different combinations of hyper parameters. Other limiting factors include limited capacity of the experimental setup, leading to stagnation in exploring the model's complete potential. SFNMT may be used for all other 22 official Indian regional languages.

REFERENCES

[1] P. Koehn, *Statistical machine translation*, vol. 9780521874151. Cambridge University Press, 2009.

[2] F. J. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2002, vol. 2002-July, pp. 295–302, doi: 10.3115/1073083.1073133.

[3] Q. Wang, J. Zhang, and C. Zong, "Synchronous Inference for Multilingual Neural Machine Translation," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 30, pp. 1827–1839, 2022, doi: 10.1109/TASLP.2022.3178241.

[4] N. J. Khan, W. Anwar, and N. Durrani, "Machine translation approaches and survey for indian languages," *arXiv*, Jan. 2017, [Online]. Available: http://arxiv.org/abs/1701.04290.

[5] H. Choudhary, A. K. Pathak, R. R. Saha, and P. Kumaraguru, "Neural machine translation for English-Tamil," in *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, 2019, pp. 770–775, doi: 10.18653/v1/w18-6459.

[6] S. Singh, M. Anand Kumar, and K. P. Soman, "Attention based English to Punjabi neural machine translation," *Journal of Intelligent and Fuzzy Systems*, vol. 34, no. 3, pp. 1551–1559, Mar. 2018, doi: 10.3233/JIFS-169450.

[7] M. Haque and M. Hasan, "English to Bengali machine translation: an analysis of semantically appropriate verbs," in *2018 International Conference on Innovations in Science, Engineering and Technology, ICISET 2018*, Oct. 2018, pp. 217–221, doi: 10.1109/ICISET.2018.8745626.

[8] M. A. Sipra, "A linguistic study of borrowings from English to urdu," *International Journal of Applied Linguistics and English Literature*, vol. 2, no. 1, pp. 203–207, Jan. 2013, doi: 10.7575/ijalel.v.2n.1p.203.

[9] K. Lingam, E. Ramalakshmi, and S. Inturi, "English to Telugu rule based machine translation system: a hybrid approach," *International Journal of Computer Applications*, vol. 101, no. 2, pp. 19–24, 2014, doi: 10.5120/17659-8474.

[10] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2015.

[11] J. Armengol-Estapé, M. R. Costa-jussà, and C. Escolano, "Enriching the transformer with linguistic factors for low-resource machine translation," in *Proceedings of the Conference Recent Advances in Natural Language Processing - Deep Learning for Natural Language Processing Methods and Applications*, 2021, pp. 73–78, doi: 10.26615/978-954-452-072-4_009.

[12] M. García-Martínez, W. Aransa, F. Bougares, and L. Barrault, "Addressing data sparsity for neural machine translation between morphologically rich languages," *Machine Translation*, vol. 34, no. 1, pp. 1–20, 2020, doi: 10.1007/s10590-019-09242-9.

[13] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in Neural Information Processing Systems*, vol. 4, no. January, pp. 3104–3112, 2014.

[14] S. A. Mohamed, M. A. Abdou, and A. A. Elsayed, "Residual Information Flow for Neural Machine Translation," *IEEE Access*, vol. 10, pp. 118313–118320, 2022, doi: 10.1109/ACCESS.2022.3220691.

[15] K. Cho *et al.*, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734, doi: 10.3115/v1/D14-1179.

[16] A. Vaswani *et al.*, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 2017-December, no. Nips, pp. 5999–6009, 2017.

[17] A. Graves, "Generating sequences with recurrent neural networks," *arxiv*, 2013, [Online]. Available: http://arxiv.org/abs/1308.0850.

[18] S. Siripragada, J. Philip, V. P. Namboodiri, and C. V. Jawahar, "A multilingual parallel corpora collection effort for Indian languages," *LREC 2020 - 12th International Conference on Language Resources and Evaluation, Conference Proceedings*, pp. 3743–3751, 2020.

[19] N. Choudhary and G. N. Jha, "Creating multilingual parallel corpora in Indian languages," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8387 LNAI, 2014, pp. 527–537.

[20] T. Kudo and J. Richardson, "SentencePiece: a simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 66–71, doi: 10.18653/v1/D18-2012.

[21] K. Wangchuk, S. C. Navaneethakrishnan, Y. Jamtsho, and Y. Wangchuk, "Dzongkha to English translation using the neural machine translation approach," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 31, no. 2, pp. 885–892, 2023, doi: 10.11591/ijeecs.v31.i2.pp885-892.

[22] D. A. Hameed, T. A. Faisal, A. K. Abbas, H. A. Ali, and G. T. Hasan, "DIA-English-Arabic neural machine translation domain: sulfur industry," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 27, no. 3, pp. 1619–1624, 2022, doi: 10.11591/ijeecs.v27.i3.pp1619-1624.

[23] M. Beseiso, S. Tripathi, B. Al-Shboul, and R. Aljadid, "Semantics based english-arabic machine translation evaluation," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 27, no. 1, pp. 189–197, 2022, doi: 10.11591/ijeecs.v27.i1.pp189-197.

[24] M. K. Nyein and K. M. Soe, "Source side pre-ordering using recurrent neural networks for English-Myanmar machine translation," *International Journal of Electrical and Computer Engineering (IJEECS)*, vol. 11, no. 5, pp. 4513–4521, 2021, doi: 10.11591/ijece.v11i5.pp4513-4521.
[25] M. A. Ayu, T. Mantoro, and J. Asian, "Quality translation enhancement using sequence knowledge and pruning in statistical machine translation," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 16, no. 2, pp. 718–727, 2018, doi: 10.12928/TELKOMNIKA.v16i2.8687.

BIOGRAPHIES OF AUTHORS

Ritesh Kumar Dwivedi he joint director (IT) in Govt. of India. He pursing Ph.D. from Department Computer Science and Engineering, Sharda University, Greater Noida, India. He has more than 18 years of experience in software application development. He has keen interest in developing AI and ML applications. He can be contacted at email: ritesh.dwivedi@nic.in.

Parma Nand professor (Dr.) Parma Nand is Pro Vice Chancellor and Professor Dept. of Computer Science and Engineering, Sharda University Greater Noida. He has over 29 years of teaching, industry and research experience. He has expertise in Wireless and Sensor Network, Cryptography, Algorithm and Computer Graphics. He has earned his Ph.D. from IIT Roorkee, M.Tech. and B.Tech. in Computer Science and Engineering from IIT Delhi. He can be contacted at email: parma.nand@sharda.ac.in.

Om Pal received B.E. in Computer Science and Engineering from Dr. B. R. Ambedkar University, Agra, MBA(O&M) from IGNOU, MS (Research) in field of Cryptography from IIT Bombay and Ph.D. in field of Cyber Security from Jamia Millia Islamia, New Delhi. Presently he is Pursuing "PG Diploma program (online mode) in Cyber Law, Cyber Crime Investigation and Digital Forensics" from National Law Institute University, Bhopal. He can be contacted at email: opal@cs.du.ac.in.