Dijlah Journal of Engineering Science (DJES) Vol. 2, No. 4, December, 2025, pp.301-315
ISSN: Printed: **3078-9656, Online: 3078-9664**, paper ID: 104

Instant Multilingual Translation using Deep Learning Techniques

Khaleel J. Hammadi[1], Nazar J. Hussain[2], Jabbar Sh. Jahlool[3]
[1]AL HIKMA University College, Baghdad, Iraq
[2,3,]Department of computer engineering techniques, College of Engineering Technologies ,University of Dijlah, Baghdad, Iraq

Article Info

Article history:

Received Oct..,24, 2025
Revised Nov., 20, 2025
Accepted Dec., 10,2025

Keywords:

Arabic Translation
Deep Learning,
Optimization
Training data
Machine Learning Algorithm

ABSTRACT

The objectives of this paper is to avoid the difficulty of analysis and the high amount of time that is required by conventional translation methods and reduce the validation loss by simply using training data when augmenting data. In addition, To make deep learning easier for researchers including the processing optimization unit and framework design And Utilize a wide variety of performance measurements, analyze how well the model are doing their jobs. The power leverage of transformers model, specifically BERTs (Bidirectional Encoders Representation from Transformer), for the task of translating text from German to English. The method used in this thesis involves pre-training BERT on extensive German text corpora, followed by fine-tuning with a focus on translation-specific data. We extend the investigation to incorporate the Marian Transformer, a dedicated neural machine translation architecture. The noteworthy outcome of our research is the achievement of a validation loss of 0.11 with the Marian Transformer. This remarkably low loss metric reflects the model's proficiency in producing highly accurate translations during validation. It underscores the effectiveness of our approach in capturing linguistic nuances and context while facilitating smooth cross-lingual communication. The results of this work is by provide 0.11 train and test loss which is better than the approaches in the related works. The importance of validation loss as a reliable indicator of translation quality. This research contributes to advancing the state-of-the-art in machine translation and opens avenues for improved multilingual communication.

Corresponding Author:

Khaleel J. Hammadi
AL HIKMA University College, Baghdad, Iraq
Yarmouk, Four Streets Intersection, Baghdad, Iraq
Email : dr.khal66@gmail.com

1. INTRODUCTION

The state of art in several artificial intelligent duties has considered in deep learning improvements such as speech recognition, object detection, and machine translation [1]. The opportunity of solve more difficult task is to develop deep architecture nature grant deep learning technologies resulting in extend variety of different modern domain [2]. The artificial intelligence has dramatically changed the basic concept of computation algorithm and evolved from niche to mainstream with many shifting aspects of societies [3]. Machine and deep learning are now move toward understanding the world around the human due to many applications that provide better muscle and physics simulation, and effective cancer detections [4]. This huge development could be achieved under deep learning which required to be fast effective in term of correctness and portable over heterogeneousness device [5].

Therefore, developer and researchers used the DL software to confirm this property and shorten the implementations of different model to offer a variety of standards model [6]. The deep learning facilitates machine learning and provide high impact on the idea of researcher. In this field, many researchers introduce different approaches. The recurrent neural network has been used by [7] to denoise speech signal under stacked auto encoder and discover cluster pattern of gene expression. By using neural model to generate image with different style, the new model has been proposed by [8]. Deep learning has been used by [9] to allow sentimentality analysis from multiple modality concurrently. These periods are witness the blooming of deep learning researches. Dramatically, the machine learning has reshaped different field of computer science like computational biology, computer visions, robotic, and natural language processing [10 -20]. It is ongoing to gain purchase in new area containing program synthesis [21]. In particular, the deep learning has ambitious development in these areas and is currently powering varied in the industries like translation, search, security, and recommendation [22]. Machine Learning is an arena in AI which is use to clarify the machine how to switch information further professionally. The machine learning is applying to understand the outlines and excerpt pertinent information from data. This idea mentions to the variations in system that performs task related to artificial intelligence involving prediction, robot control, diagnosis, recognition, and planning [23].

Through the years, machine learning algorithm has progress from chastely statistics model to bio-inspired mode such as neural networks. To discriminate them is classical or traditional ML model and DL [25]. Figure 1 shows the practice ML technique as partial to process of normal rows data. It is required a important domain expert to design the appropriate features extractors that will transforms rows data to the right arrangement from classifier that detected the patterns to classify the inputs. This problem could be solved by Deep Learning which ones are valued to the classifications [26].

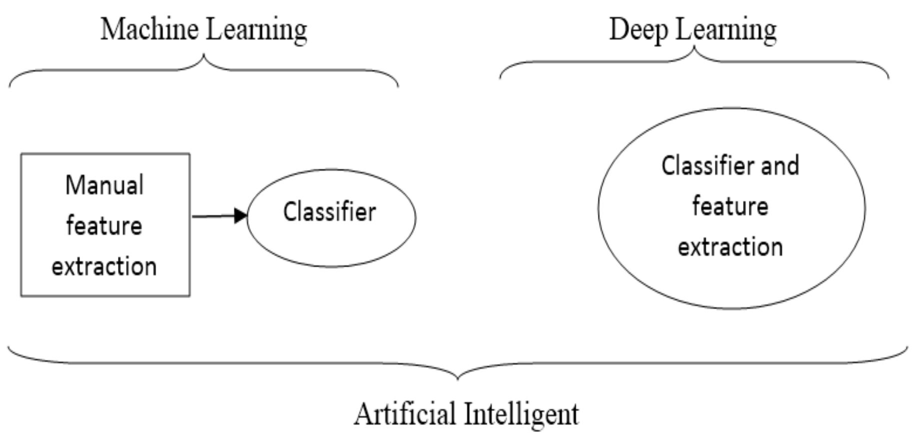


Figure 1: Machine Learning (ML) vs Deep learning (DL)

Automatically, DL is a ML method that clarify computer to do what come logically by used a bio-inspired construction called deep neural network [27]. This construction permits the machine to feed with rows data and robotically abstract the symbol need for the wanted duty. It is the knowledge overdue a lot of stun application such as language Translation, driverless car, and even some hospital that use prognostic application to accomplish patients on the emergency waiting room [28-36]. This paper introduces clear deep learning tools and state of art with a comparative study framework rendering to different criterial. An overall obtainable structures supports many DL operator, popularities, hardware design, and size of community. Forward pass time is considered to measure the performance as prediction from model to know which deep learning tools will provide the user demand with superlative environments to test the model. This paper consists of the following sections: section one includes the materials and methods used in the experiment part, section two contain the results discussion, and finally the conclusion of this work.

2. MATERIALS AND METHODS

There are many stages that used to implement our work. The hand-craft feature were mainly use to models usual languages task until deep learning method derived everywhere and solve some of the problem face by outdate ML model like the curse of dimensional. We will explain the most significant word representation techniques, which include converting words from the vocabulary into vectors or numbers.

Distribution vector, also named word implanting is bases on the so-named distribution hypothesis — words displaying with same contexts have same meanings. Word embedding are pre-train on tasks where the aim is to predicts words base on its contexts, naturally using shallows neural networks. In the past, Distributed representations were significantly employed to study numerous natural language processing tasks, but they only managed to become more well-known when the skip-gram models and continuous bag-of-words (CBOW) were introduced to the domain. They were trendy because they could be applied for semantic compositionality (e.g., 'man' + 'royal' = 'king') and might professionally create higher-qualities words embedding as show in Figure 2.

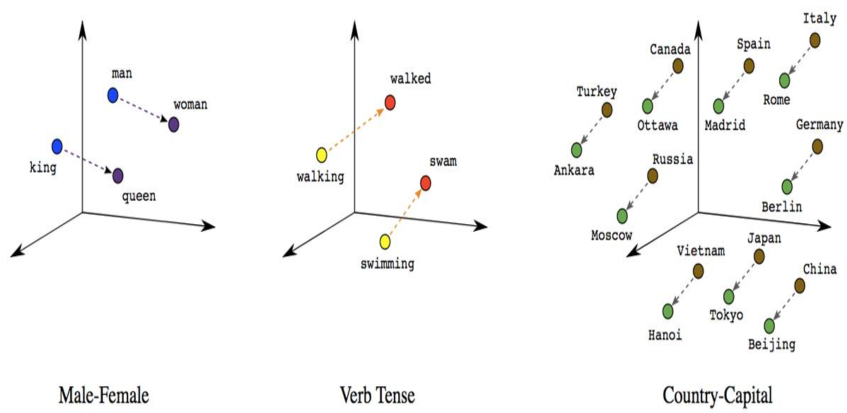


Figure 2: word embedding

The issue with word embedding approaches is at the time we intend to get vector representations for the words “hot potato” or “Boston Globe”. Since these words don't represent the mixture of the definitions of the specific words, we can't basically join the vector representations of the specific words. When taking longer sentences and phrases into account, it gets even harder. Another disadvantage of words embedding is that it depends on the applications in which it is use. Although this is often computationally expensive, it has been investigated to retrain tasks-specific embedding for every newer task.

The goal of CBOW, a neural approach to building word embedding, is to determine the conditional probability of a target word provided the context words in a specific window size. On the other side, the goal of Skip-gram, a neural method for constructing word embedding, is to predict the surrounding context words (i.e., conditional probability) given a central target word. The word embedding dimension for both models is calculated (using an unsupervised method) by the accuracy of the prediction.

One disadvantage of the word2vec models is that using shorter window sizes results in similar embedding for opposing words like "bad" and "good" which is undesirable, particularly for tasks like sentiment analysis in which this distinction is crucial. Other issues with Word2vec models include their failure to take into account polysemy and other biases that might be found in the train data.

In contrast to Word2Vec, which generates word representations that are static and unaffected by the context in which they appear, BERT generates word representations that are dynamically changed by the surrounding words. As an illustration, consider the following two sentences: "The suspect was charged with robbing a bank.". " By the river bank, a man started hunting."

Word2Vec would create the same word embedding for the word "bank" in both sentences, whereas BERT would create a different word embedding for "bank" in each phrase. Better model performance is attained as a result of the context-informed word embeddings' capacity to collect additional sorts of information that result in more accurate feature representations, in addition to catching obvious differences like polysemy.

2.1 Sequence Models

Sequence models are neural networks that input or output sequences of data. Sequential data contains audio clips, video clips, text streams, time-series data and etc. The most common neural networks in deep learning are

Long short-term memory (LSTM) and recurrent neural networks (RNNs). Transformers and Long Short-Term Memory (LSTM) networks are two popular and highly effective models in domains of normal languages process and machines translation applications. Despite the fact that both approaches have been successful in a number of applications, their architectural structures and approaches to handling and processing data are different.

2.2 Recurrent Neural Network (RNN)

RNN is a generalization of feed forward neural network that has an interior memorial as illustrated in Figure 3. RNN is recurrent in natural since they perform similar functions for each data inputs, and the outcomes of the present inputs depend on the outcomes of the prior computations. The outputs are produce, copy, and then return to the recurrent networks. In case of make a decision, it considers both the current inputs and the outputs that it has learnt from the prior inputs. Unlike feed forward neural networks, RNNs can process input sequences utilizing their internal state (memory). As a result, they can be applied to tasks like speech recognition or unsegment, connected handwriting recognition. Other neural networks' inputs all function independently of one another. However, every input in an RNN is linked to every other input.

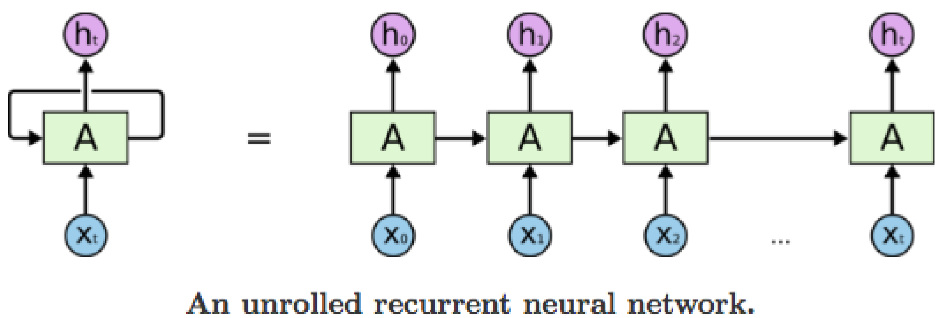


Figure 3: RNN Architecture

The first thing is to extract the X(0) from the structure of input, after which it output h(0), which, along with X(1), serve as the inputs for the subsequent steps. Hence, the inputs for the subsequent steps are h(0) and X(1). Similarly, h(1) from the succeeding steps are the inputs with X(2) for the subsequent steps, and so on. In this mean, it continue to remember the contexts through training.

The equation to determine the present state is:

$$h_t = f(h_{t-1}, x_t) \tag{1}$$

Applying Activation Function:

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}X_t) \tag{2}$$

Where
W: weights
H: single hidden vectors
W_{hh}: weights at previous hidden states
$W_{xh:}$ weights at present inputs states
t_{anh}: Activation functions that implements a non-linearity that squashes the activations to the range [-1,1].
Then the output is:

$$y_t = W_{hy}h_t \tag{3}$$

where
Y_t: outputs state
W_{hy} : is weights at outputs state

2.3 Types Of RNN

There are four types of RNNs depending on the network's number of inputs and outputs as show in Figure 4.

1. One to One
2. One to Many
3. Many to One
4. Many to Many

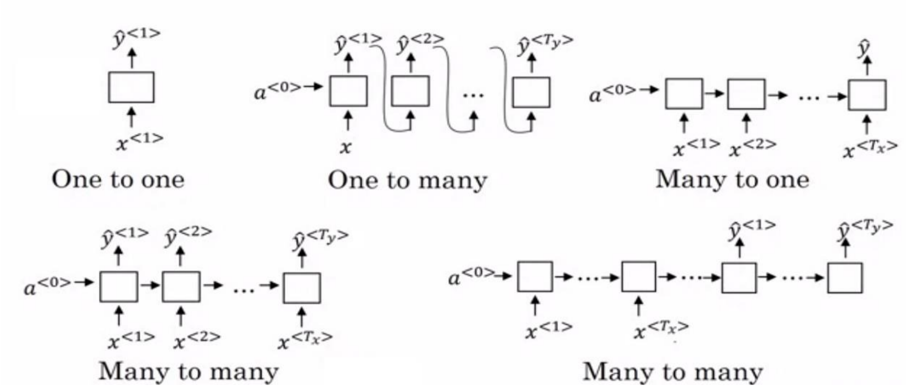


Figure 4: Types Of RNN

Many-to-Many RNN (Tx>1,Ty>1) Architecture, as is obvious, accepts numerous inputs and produces multiple outputs, but many-to-many models can be two types, as seen above:

1. Tx=Ty

In this case, the Input and output layers have the same size. This can also be thought of as every input having an output, and named entity recognition is a popular application.

2. Tx!=Ty

Machine translation is where this kind of RNN architecture is most frequently used. Models with varying input and output layer sizes can also represent many-to-many architecture.

For example, the three lovely English word "I Love You" is reduced to the two words "te amo" in Spanish. Machine translation models use non-equal Many-to-Many RNN architecture in the background, which enables them to return words that are either more or less than the input string.

2.4 Vanishing and Exploding Gradients

Let's first explore the gradient

Gradient: A gradient is a partial derivative in relation to its inputs. A gradient calculates how much functions outputs will diverges if its input is somewhat changed. The gradients can alternatively be thought of as a function's slope. The greater the gradient, the steeper the slope, and the more quickly a model can learn. The models stop learning if the slope is nearly zero. The gradient basically calculates the difference in wholly weight relatives to the errors change.

Gradient problems in RNN

The gradient can occasionally get too tiny or too big while an RNN algorithm is being trained. As a result, in this situation, training an RNN algorithm becomes quite challenging. Due to this, the problems include long training period, low accuracy, and poor performance will occur.

Exploding Gradient

Exploding gradient problem happens when we assign the weights a high priority. In this situation, gradient values become too huge, and the slope essentially grows exponentially. To resolve this, the techniques include identity initialization, truncated back-propagation, and gradient clipping could be used.

Vanishing Gradient

When a gradient's values are too small, the problem arises and the model either stops learning or learns very slowly. To resolve this, the techniques include weight initialization, choosing the right activation function, and the best approach to deal with the vanishing gradient problem is the use of LSTM (Long Short-Term Memory) could be used.

The benefits of RNN is:

1. RNN could modeled the structure of data so that every samples could be assume to be dependents on preceding one.
2. RNN is even use with convolution layer to extends the active pixel neighborhood.

The drawback of RNN is:

1. Gradients vanishing and exploding problem.
2. It's really challenging to train an RNN.
3. It can not processing long sequence when use t_{anh} or relu as an activation functions.

LSTM

Long Short-Term Memory (LSTM) network is an altered versions of RNN, this make it humbler to evoke previous information from memories. The RNN's vanishing gradients problems is solved here. LSTMs is well-suite to classifying, processing, and predicting time sequence given time lag of unidentified durations. The model is trained via back-propagation. In an LSTM network, three gates are introduced as show in Figure 5.

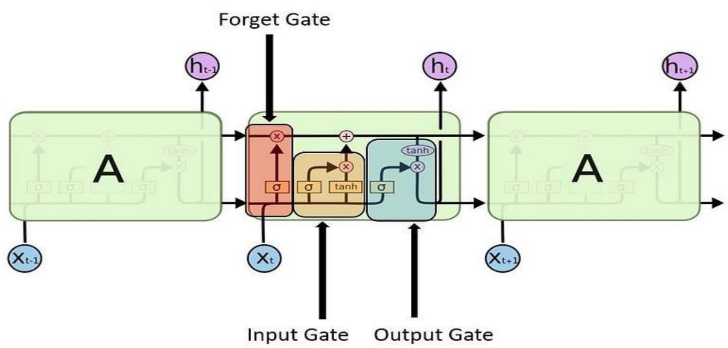


Figure 5: LSTM Architecture

a. **Inputs gates:** define which inputs values should be applied to the memories modification. The sigmoid functions specifies which numbers to pass 0,1. The t_{anh} functions assign weights to the value that are provided, determine their importance on a scale of -1 to 1

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (4)$$

$$\tilde{C}_t = tanh\ tanh\ (W_C \cdot [h_{t-1}, x_t] + b_C) \quad (5)$$

b. **Forget gates:** define which data must be remove from the blocks. A sigmoid functions make that decision. Every numbers in the cells states Ct-1, it output a numbers among 0 (omit this) and 1 (keep this) by using the previous state (ht-1) and the contents inputs (Xt).

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \quad (6)$$

c. **Outputs gates:** The block memory and input is use to determined the outputs. The sigmoid functions determines which numbers to pass 0,1. The t_{anh} functions weighs the value that are supplied, determining their levels of importance from -1 to 1, and multiply by the outputs of Sigmoid.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (7)$$

$$h_t = o_t * tanh\ tanh\ (C_t) \quad (8)$$

Compared to typical RNNs, LSTMs offer a more reliable and efficient method of managing sequence data because of their distinctive architecture and gating mechanisms. They are appropriate for a variety of applications, including nature languages process, machine translation, time series predicting, and texts generations, since they can captured long-terms dependency and solve the threatened gradient problems. Even though LSTM is useful, they are not the only option for dealing with series data. More recent model, such as transformer, have appeared to offer alternatives for capture long-range dependency. They have become more popular due to the way they handle long-range context and their enhanced performance in machine translation task.

2.5 Transformer

Transformers shown in Figure 6 have gained approval in current time because of the introduction of the attention mechanism and their parallelization capabilities. This enables them to analyze huge, complicated sequences efficiently without becoming slowed down by sequential data processing. They manage to achieve this without the use of recurrent neural networks (RNNs) and LSTMs. This enables parallel processing, which leads to shorter training durations than with RNNs' sequential methods. In applications including text classification, speech recognition, and machine translation, transformers have shown excellent performance. Stacks of encoders and decoders layer with layers of feed-forward neural networks (FFN) and self-attention make up the architecture of transformers.

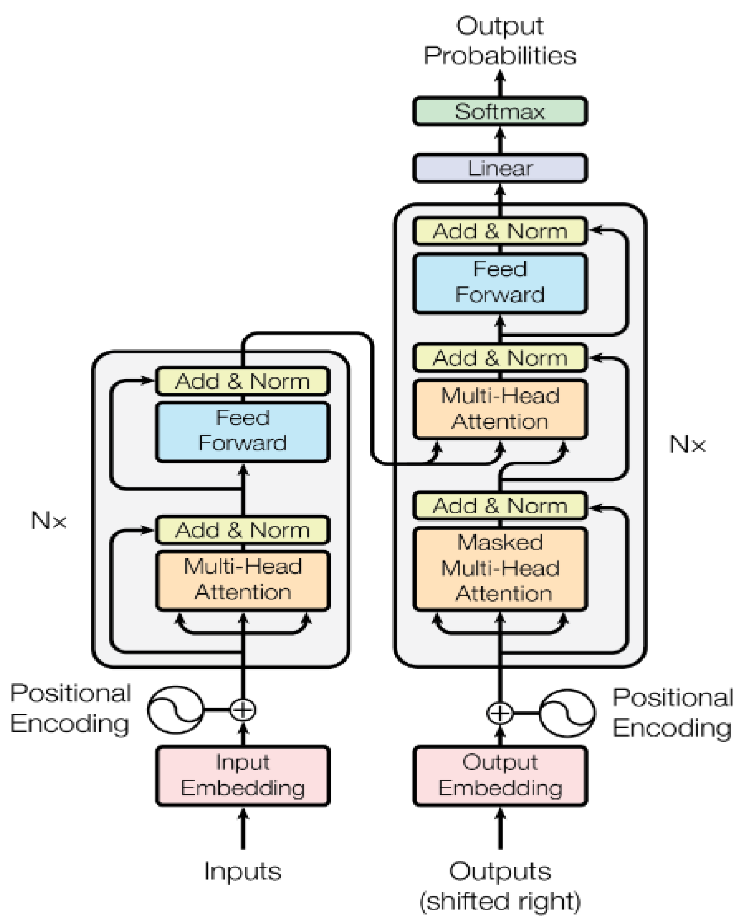


Figure 6: Transformer Architecture

3. CONSIDERATION MECHANSIM

The Consideration Mechanism show in Figure 7 is a key development in neural network that enables models to concentrate on specific input data subsets rather than processing the entire set at once. This has shown to be particularly useful for language translation and sequence-to-sequence tasks. The Attention Mechanism has enhanced LSTM models and opened the path for advanced network designs like Transformers.

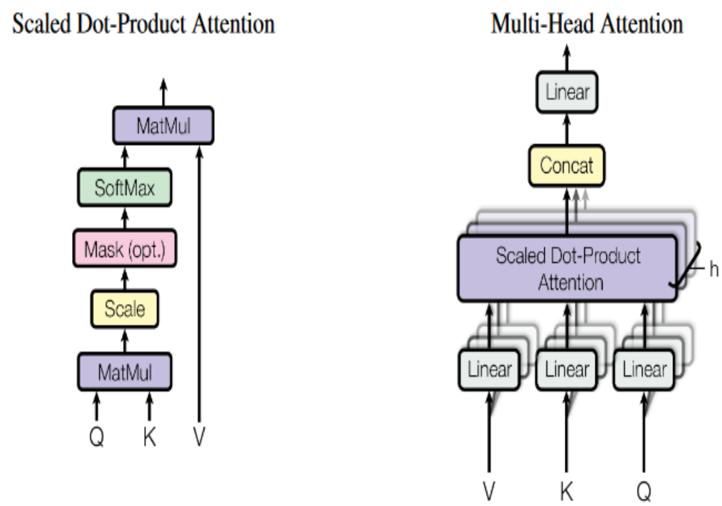


Figure 7: Attention Mechanism

a. Self-Attentions

Self-Attentions are particular kind of attentions mechanisms in which the models learn to concentrate just on specific components of the inputs series in order to provide more appropriate output. The input values are computed as a weighted sum, with the weights determined by comparing each input to the other input in the series. As a result, The model can automatically learn about the dependencies and relationships among the sequence's elements.

Self-Attention's primary building blocks are queries, values, and keys. The input elements are compared using the queries, and the relationships between the elements are represented by the keys and values. The computed attention weights are then applied to the softmax function to generate a probability distribution that highlights the sequence's most important elements.

b. Multi-Head Attentions

Multi-head attention is used in the Transformer model to concurrently concentrate on several parts of the input data, allowing the model to simultaneously learn a variety of richly contextualized data representations. The Multi-Heads attentions mechanisms has numerous attentions head, each with individual key, values, and queries, as opposed to utilizing a single attention mechanism.

The Transformer is more effective and powerful at handling complicated tasks due to this design, which allows it to store different aspects of the input sequence. Each head independently processes the input, combining representations that produced via a linear transformation and concatenation.

c. Encoder-Decoder Attention

Another essential component of the attention process is encoder-decoder attention as show in Figure 8. Mostly used in sequence-to-sequence tasks like machine translation. The input sequence is processed by the encoder to create a context vector, which the decoder then utilizes to create the output sequence.

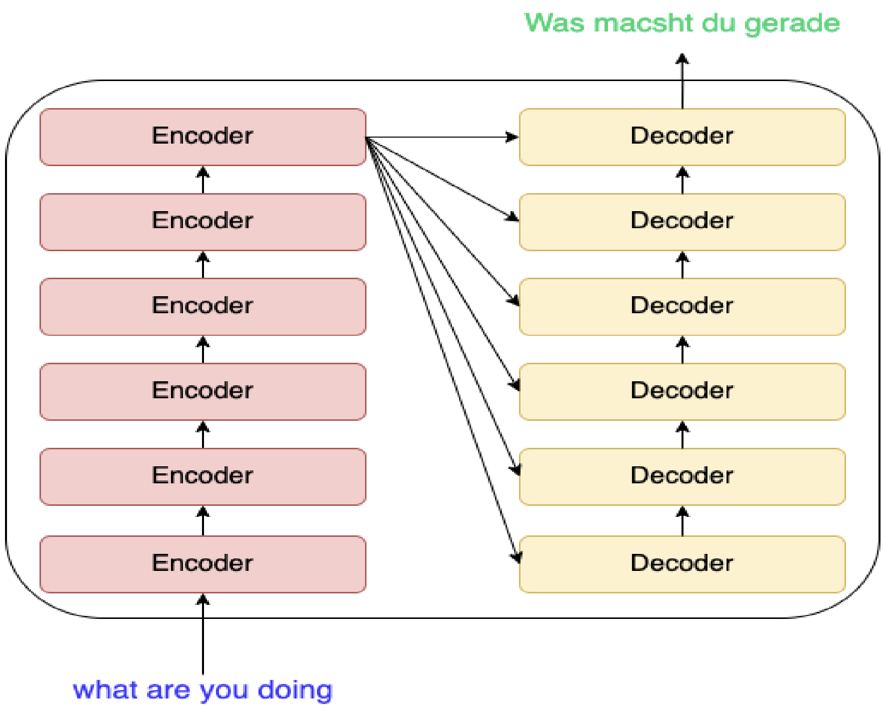


Figure 8: Encoder-Decoder Attention

The Encoder-Decoder Attention mechanism allows the decoder to concentrate on various aspects of the encoded input sequence, promoting a better knowledge of the connections between the inputs and producing output sequences with higher accuracy. The keys and values come from the encoder's output, while the queries come from the hidden states of the decoder. When generating the output, this design efficiently enable the decoders to align itself to various sections of the inputs sequences, improving translation and sequence generation.

d. Decoding and Encoding

The transformers models are made up of encoders and decoders. The two fundamental building elements of each layer in both the encoder and decoder are the position-wise feed-forward network and multi-head self-attentions.

Encoder: To create a continuous representation, the encoder processes the input sequence after taking it in. This continuous representation efficiently enables the decoder to produce the desired sequence while maintaining the contextual information of the input.

Decoder: The decoder creates the target sequence using the continuous representation provided by the encoder. In addition to having an encoder-decoder attention mechanism that enables it to concentrate on various segments of the inputs sequences, it also contains a multi-head self-attentions mechanism.

e. Positional Encoding

Transformers need a way to preserve the sequence's order because they lack recurrent structures. Positional encoding is useful in this situation. The input embedding are enhanced with positional encoding to give the model positional data. Usually, the input embedding are modified by combining cosine and sine functions of various frequencies. These functions support the model's efficient collection and application of the positions of the input tokens. This function enhance the model efficient learning and the usage of the positions of the input tokens.

3.1 Residual Connections

The crucial component of the transformer construction is residual connections. They help to solve the vanishing gradient issue and allow the model to maintain data from previous layers. In transformers, for each sub-layer (position-wise feed-forward networks and multi-head self-attention), there is a residual connection followed by a layer normalization process. As a result, the input and output of each sub-layer are summed, and the resulting sum is then normalize before being sent on to the following sub-layer.

3.2 Parallelization

The main advantage of the transformer models compared to RNN and LSTM is the capability of processed the inputs series in parallel rather than successively. Transformer used self-attentions mechanism, which might processing numerous word concurrently, as opposed to recurrent connections, which process input sequentially. Transformers are excellent at handling large-scale natural language processing tasks because of their ability to perform parallel computing.

3.3 Handling Long-Ranges Dependency

Because of the self-attentions mechanism, Transformers handle long-range dependencies more effectively than LSTMs, which is another significant advantage. Due to their ability to maintain information from distant regions in longer sequences, they can weighs the reputation of many position in the inputs series in contrast to LSTM.

3.4 Training Time

Transformers and LSTMs are two commonly used techniques for training neural network in DL task. Their training varies in duration. For better parallelization, transformers are known to train faster than LSTMs. Transformers use the self-attention mechanism, which, in contrast to LSTMs, does not rely on sequential calculations, explaining how this is applicable. In addition, better utilization of current GPU architectures due to this parallelization might accelerate the training process.

4. DATASET

We used a machine translation dataset for English and German languages that contains sentence pairs from the Tatoeba Project [1]. The dataset contains 221532 sentence pairs of English and German languages. We displayed the distribution of German and English sentence length as show in Figure 9.

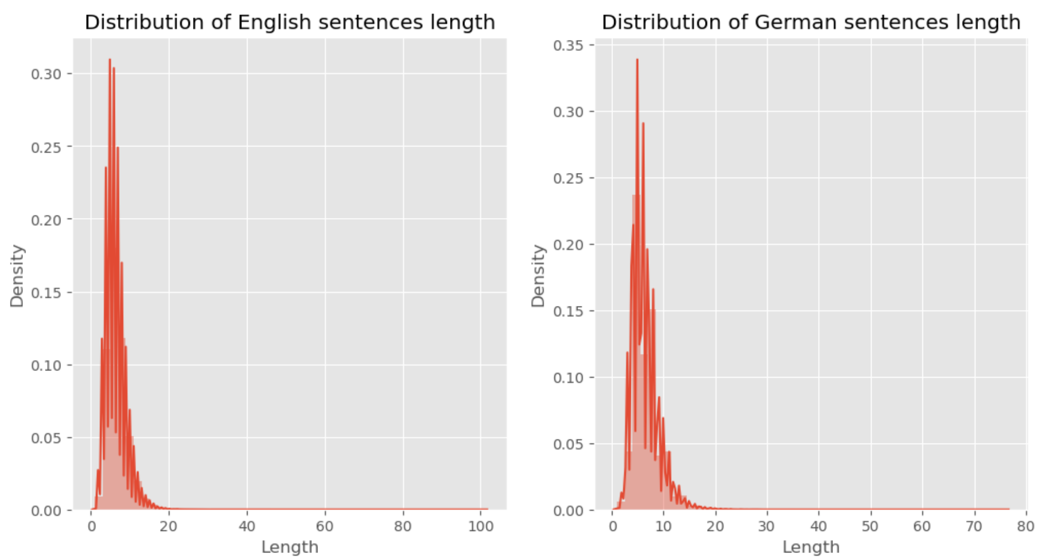


Figure 9: the distribution of German and English sentence length

Figure 10 and Figure 11 display the most frequent words in English and German sentences as word cloud. Word clouds are graphic representations of words that give words with more frequent occurrences greater attention. The most frequently written words will dynamically grow in size.

Figure 10: Word Cloud for the most frequent words in English sentences

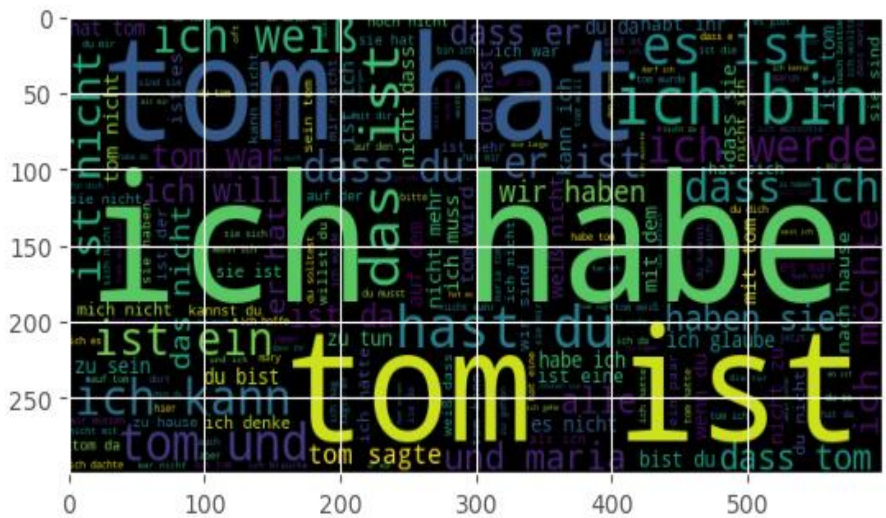


Figure 11: Word Cloud for the most frequent words in German sentences

4.1 Data Preprocessing

We converted all English and German sentences to lowercase and split the dataset into training-set and test-set with a test-set size of 20%. The training set contains 177225 sentence pairs of English and German languages and the test set contains 44307 sentence pairs of English and German languages. After that, we tokenized all the sentences using a Bert-base-German-cased tokenizer with max input and target length equal to 105 which converts words from his own vocabulary into numbers based on the BERT Embedding concept mentioned above. The BERT tokenizer had a brilliant idea to use the word-piece tokenizer concept, which simply divides some words into sub-words. For instance, if the word "sleeping" is not found in the tokenizer's vocabulary, then it is tokenized into the words "sleep" and "##ing". Using this strategy is often helpful in separating unknown words into familiar ones.

4.2 Training Process

We trained a transformer called the Marian transformer which consists of 6 layers in each component of the encoder and decoder. The Marian transformer is smaller than many other machine translation models which makes it a light model with a small size and one of the fastest models when it comes to the training phase. It is well-known for the machine translation tasks. The parameters for the training process shown in Table (1) as Following:

Table (1): Training parameters

Parameters	**Values**
Learning Rate	0.00005
Batch Size	64
Epochs	10
Weight Decay	0.01
Loss Function	Sparse Categorical Cross entropy
Training Time	5 Hours and 6 Minutes
Accelerator	GPU P100

These are the best parameters we found after trying many experiments on the dataset and they achieved the best results among all other results. Now we have a great model that can convert any German sentence to its English counterpart like this German sentence show in Figure 12.

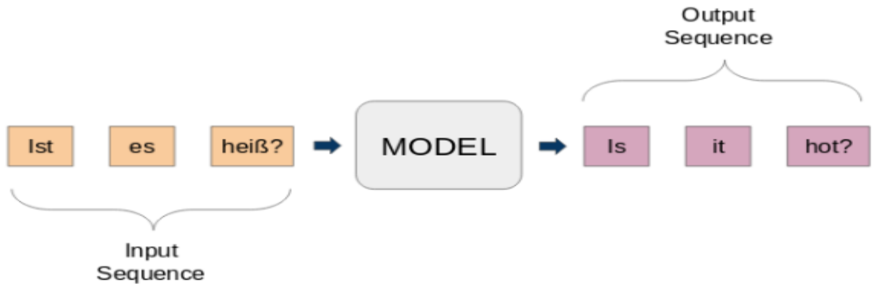


Figure 12: Machine Translation using the Model

5 DISCUSSION

Finally, Our powerful model stands as a testament to the ever-evolving landscape of artificial intelligence, promising seamless language translation without errors or inaccuracies. In educational realms, envision students accessing a world of knowledge regardless of language barriers, promoting cross-cultural understanding and collaborative learning. Similarly, within the business sphere, this innovation fosters international partnerships, breaking down linguistic obstacles that often hinder global transactions. Delving into the technical intricacies, such as dataset enhancement and result optimization, can illuminate the depth of our contribution, showcasing how advancements in machine translation are reshaping the way we perceive and engage with language. Such discussions not only highlight the significance of our work but also underscore the vital role technology plays in creating a more interconnected and inclusive global society.

6 CONCLUSION

This introduces the deep learning outline and compliers with optimized techniques to overcome the challenging application which is successfully proved the efficiency. There is a rising demand to test different field application and deploy smart and wide spectrums of devices. In deep learning, the challenge that faced the software improvement is multiple include workload and device variety. This processing is complex by diversity of hardware specifications include embedded ASICs, FPGA, GPU, and CPU. In addition, the model and operator's optimization time to train could be ranged from hours to week which cannot permit to developer and researcher to test different models. Whilst, the time to infer resulting from model range or from few milliseconds in simple neural network to complex one that seen slow for end user. Consequently, the researcher among the community tackle the challenges of optimized model by innovating in term of new operator or by execute the bunching of optimized code. The presents a prune methods to reduced the numbers of connection of deeps neural network without affected accuracies. Our prune methods, motivation in part by how learn work in the humans brains, operate by learn which connection are important, prune the un-important connection, and then retrain the remaining sparse networks. This thesis highlight the powers of suggested prune methods through experiment which reveal that full connection layer and convolutional layer could be prune the numbers of connection of convolutional layer is reduced by 3x and full connect layer by 10x with out losses of accuracies. With the Neural Talk experiments on Flickr-8K, we found LSTM could also be prune by 10x. Prune reduce the amounts of computations require by deeps neural network. The numbers of potentials direction for future work include hardware compiler with users' interface and optimizations for specifics application, like remote sensing or medicals images analyzing, is one. Additional is the investigations of IR appropriate for hardware generations to accelerating both inferences and train. In addition, it is to devised active techniques for the explorations of designs space of many data flows graphs-base DNNs model, include for existing network.

REFERENCES

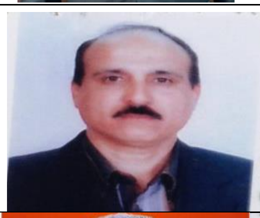
[1] Byungsoo Kim et al., « Lagrangian Neural Style Transfer for Fluids », ACM Transactions on Graphics 39.4 (2020)
[2] Scott Mayer McKinney et al., « International evaluation of an AI system for breast cancer screening », Nature 577.7788 (2020)
[3] Lindsay, G. Convolutional neural networks as a model of the visual system: Past, present, and future. J. Cogn. Neurosci. 1–15 (2020).
[4] Lee, W., Kim, D., Hong, S. & Lee, H. High-fidelity synthesis with disentangled representation. arxiv (2020).
[5] Chang, L., Egger, B., Vetter, T. & Tsao, D. Y. What computational model provides the best explanation of face representations in the primate brain? bioRxiv (2020).
[6] Duan, S. et al. Unsupervised model selection for variational disentangled representation learning. ICLR (2020).
[7] Abdul Rauf, S. and Yvon, F. (2024) 'Translating scientific abstracts in the bio-medical domain with structure-aware models', Comput. Speech Lang., Vol. 87, DOI: 10.1016/j.csl.2024.101623.
[8] Ahda, F.A., Wibawa, A.P., Prasetya, D.D. and Sulistyo, D.A. (2024) 'Comparison of Adam optimization and RMSprop in Minangkabau-Indonesian bidirectional translation with neural machine translation', International Journal on Informatics Visualization, Vol. 8, No. 1, DOI: 10.62527/joiv.8.1.1818.
[9] An, Z., Wu, J., Yang, M., Zhou, D.L. and Zeng, B. (2024) 'Unified quantum state tomography and Hamiltonian learning: A language-translation-like approach for quantum systems', Phys. Rev. Appl., Vol. 21, No. 1, DOI: 10.1103/PhysRevApplied.21.014037.
[10] Anderson, R., Scala, C., Samuel, J., Kumar, V. and Jain, P. (2024) 'Are emotions conveyed across machine translations? Establishing an analytical process for the effectiveness of multilingual sentiment analysis with Italian text', Journal of Big Data and Artificial Intelligence, Vol. 2, No. 1, DOI: 10.54116/jbdai.v2i1.30.
[11] Belete, M.D., Salau, A.O., Alitasb, G.K. and Bezabh, T. (2024) 'Contextual word disambiguates of Ge'ez language with homophonic using machine learning', Ampersand, Vol. 12, DOI:10.1016/j.amper.2024.100169.
[12] Chandrakala, C.B., Bhardwaj, R. and Pujari, C. (2024) 'An intent recognition pipeline for conversational AI', International Journal of Information Technology, Vol. 16, No. 2, Singapore,DOI:10.1007/s41870-023-016428.
[13] Guimarães, N., Campos, R. and Jorge, A. (2024) 'Pre-trained language models: What do they know?', Wiley Interdiscip Rev. Data Min. Knowl. Discov., Vol. 14, No. 1, DOI: 10.1002/ widm.1518.

[14] Han, L., Gladkoff, S., Erofeev, G., Sorokina, I., Galiano, B. and Nenadic, G. (2024) ‘Neural machine translation of clinical text: an empirical investigation into multilingual pre-trained language models and transfer-learning’, Front. Digit Health, Vol. 6, DOI: 10.3389/fdgth. 2024.1211564.
[15] Jing, X. (2024) ‘Automatic recognition of machine English translation errors using fuzzy set algorithm’, Soft Comput., DOI: 10.1007/s00500-023-09543-5.
[16] Ju, Z., Xin, Y. and Ye, M. (2024) ‘Machine translation of classical Chinese based on unigram segmentation transformer framework’, Applied and Computational Engineering, Vol. 37, No. 1, DOI: 10.54254/2755-2721/37/20230465.
[17] Karabayeva, I. and Kalizhanova, A. (2024) ‘Evaluating machine translation of literature through rhetorical analysis’, Journal of Translation and Language Studies, Vol. 5, No. 1, DOI: 10.48185/jtls.v5i1.962.
[18] Kaur, K. and Chauhan, S. (2024) ‘A comparative analysis of lexical-based automatic evaluation metrics for different Indic language pairs’, Journal of Autonomous Intelligence, Vol. 7, No. 4, DOI: 10.32629/jai.v7i4.1393.
[19] Kjell, O.N.E., Kjell, K. and Schwartz, H.A. (2024) ‘Beyond rating scales: With targeted evaluation, large language models are poised for psychological assessment’, DOI: 10.1016/j.psychres.2023.115667.
[20] Luo, S., Ivison, H., Han, S.C. and Poon, J. (2024) ‘Local interpretations for explainable natural language processing: a survey’, ACM Comput. Surv., Vol. 56, No. 9, DOI: 10.1145/3649450.
[21] Macas, M., Wu, C. and Fuertes, W. (2024) ‘Adversarial examples: a survey of attacks and defenses in deep learning-enabled cybersecurity systems’, DOI: 10.1016/j.eswa.2023.122223.
[22] Mahdi, M.G., Sleem, A. and Elhenawy, I. (2024) ‘Deep learning algorithms for Arabic optical character recognition: a survey’, Multicriteria Algorithms with Applications, Vol. 2, DOI: 10.61356/j.mawa.2024.26861.
[23] Mi, C. and Xie, S. (2024) ‘Language relatedness evaluation for multilingual neural machine translation’, Neurocomputing, Vol. 570, DOI: 10.1016/j.neucom.2023.127115.
[24] Nam, W. and Jang, B. (2024) ‘A survey on multimodal bidirectional machine learning translation of image and natural language processing’, DOI: 10.1016/j.eswa.2023.121168.
[25] Ni, P., Okhrati, R., Guan, S. and Chang, V. (2024) ‘Knowledge graph and deep learning-based text-to-graphql model for intelligent medical consultation chatbot’, Information Systems Frontiers, Vol. 26, No. 1, DOI: 10.1007/s10796-022-10295-0.
[26] Oliveira, E.E., Rodrigues, M., Pereira, J.P., Lopes, A.M., Mestric, I.I. and Bjelogrlic, S. (2024) ‘Unlabeled learning algorithms and operations: overview and future trends in defense sector’, Artif. Intell. Rev., Vol. 57, No. 3, DOI: 10.1007/s10462-023-10692-0.
[27] Patle, R. and Kalra, G. (2024) ‘enhancing natural language processing (NLP) through VIKOR method: a comprehensive approach for improved computational linguistics’, Computer Science, Engineering and Technology, Vol. 2, No. 1, DOI: 10.46632/cset/2/1/4.
[28] Pradhan, A. and Yajnik, A. (2024) ‘Parts-of-speech tagging of Nepali texts with Bidirectional LStM, conditional random fields and HMM’, Multimed. Tools Appl., Vol. 83, No. 4, DOI: 10.1007/s11042-023-15679-1.
[29] Shang, F. and Li, Y. (2024) ‘Exploring english long sentence translation methods by applying natural language processing techniques’, Applied Mathematics and Nonlinear Sciences, Vol. 9, No. 1, DOI: 10.2478/amns.2023.2.01352.
[30] Shenbagaraj, A. and Iyer, S. (2024) ‘An intelligent system for automated translation of videos from English to native language applying artificial intelligence techniques for adaptive elearning’, International Journal of Intelligent Systems and Applications in Engineering, Vol. 12, No. 3, p.2024.
[31] Sohrab, M.G., Asada, M., Rikters, M. and Miwa, M. (2024) ‘BERT-NAR-BERT: a non-autoregressive pre-trained sequence-to-sequence model leveraging BERT checkpoints’, IEEE Access, Vol. 12, DOI: 10.1109/ACCESS.2023.3346952.
[32] Sunar, A.S. and Khalid, M.S. (2024) ‘Natural language processing of student’s feedback to instructors: a systematic review’, IEEE Transactions on Learning Technologies, Vol. 17, DOI: 10.1109/TLT.2023.3330531.
[33] Wei, C., Wang, Y-C., Wang, B. and Kuo, C-C.J. (2024) ‘An overview of language models: recent developments and outlook’, APSIPA Trans. Signal. Inf. Process, Vol. 13, No. 2, DOI: 10. 1561/116.00000010.
[34] Zemni, B., Zitouni, M., Bouhadiba, F. and Almutairi, M. (2024) ‘A comparative study of the lexical ambiguity of Arabic, English, and French in natural language processing’, Journal of Intercultural Communication, Vol. 24, No. 1, DOI: 10.36923/jicc.v24i1.171.

ISSN: Printed: **3078-9656, Online: 3078-9664**, paper ID: 104

[35] Zemni, B., Zitouni, M., Bouhadiba, F. and Almutairi, M. (2024) ‘On ambiguity in the arabic language: scrutinizing translation issues through machine translation from English and French into Arabic’, Journal of Intercultural Communication, DOI: 10.36923/jicc.v24i1.171.
[36] Zhang, W., Bao, Y. and Xu, W. (2024) ‘Design and implementation of natural language processing machine translation system based on Seq2Seq model’, in Frontiers in Artificial Intelligence and Applications, DOI: 10.3233/FAIA231215.

BIOGRAPHIES OF AUTHORS

	Dr. Khaleel J.H`ammadi has been a full-time lecturer in the college of Electronic and electronic Techniques- Baghdad, Ministry of high education and scientific research/Iraq. Received his MS and PhD Degree in Electrical engineering from University of Belgrade/ Yugoslavia /1984 and University since Malaysia –Malaysia /2011 respectively. Emaile : dr.khal66@gmail.com
	**Dr. Nazar Jabbar Hussain**, Received His Msc. degree in the in Electronic Engineering from University of Technology Baghdad – Iraq in 2005 and Doctor of philosophy in video encryption and compression from University of Buckingham UK -2016. He has been a full-time lecturer in computer engineering techniques Department/ Dijlah University College, Baghdad, Iraq, since December 2018. He also worked as senior researcher in the Iraqi Center of Development and Research since 1994., Currently, he has a head of computer engineering techniques Department/ Dijlah University College/ Baghdad/ Iraq since 2017. It can be contacted at email: nazar.jabar@duc.edu.iq
	**Jabbar Shatti Jahlool** was born in Iraq in 1964. He received the B.Sc. and M.Sc. degree in Electrical Engineering from University of Technology, Baghdad, Iraq, in 1988 and 2001 respectively. From 2001 to 2014, he was electronic hardware designer, director of the technology transfer and scientific research in the Ministry of Industry. Since 2015 he has been a university lecturer with Department of Computer Techniques Engineering, Dijlah University College, Baghdad, Iraq. His research interests include hardware electrical and electronic circuits design, FPGA design, microcontroller’s research and design project, power system analysis and control, and power electronics circuits. He can be contacted at email: jabbar.shatti@duc.edu.iq.