A Transformer-Enhanced System to Reverse Dictionary Technology

Ahmed Bahaaulddin A. Alwahhab[1,*], Vian Sabeeh[1], Ali Sami Al-Itbi[1], Ali Abdulmunim Ibrahim Al-kharaz[1]
[1]Department of informatics, Technical College of Management, Middle Technical University, Baghdad, Iraq
Emails: ahmedbahaaulddin@mtu.edu.iq; viantalal@mtu.edu.iq; ali.sami@mtu.edu.iq; ali.al-kharaz@mtu.edu.iq

Abstract

The ability to retrieve a word from the cusp of memory often encounters the well-documented Tip-of-the-Tongue (TOT) barrier. This cognitive phenomenon can impede communication and learning. Addressing this, our study introduces a novel reverse dictionary framework empowered by cutting-edge neural network architectures to facilitate the retrieval of words from definitions or descriptions. This research draws the path of the development and the efficiency of various natural language deep learning models formulated to grasp the semantics inside the text. This work started with gripping a new dataset with rich content from a linguistic perspective. An accurate pre-processing step, including text normalizations and contextual features extraction, was conducted to transform the unstructured text into structured features fitting the model training. Dense vectors representative of text have been extracted using the BERT embedding model. Three models (LSTM, FNN, and GRU) were tested and compared using scrapped and benchmarked data. The proposed model that was consisted from Bert embedding and LSTM learner was evaluated and showed notable performance under cosine similarity and mean square error metrics. The LSTM model proved useful in real-world applications by exhibiting excellent semantic coherence in its embedding and accuracy in its predictions. This research evolved a discussion about the efficient behavior of the pre-trained BERT model in enhancing vocabulary. In addition, this work sheds light on the crucial role of reverse dictionaries in many NLP applications in the future. Subsequent research endeavors will focus on augmenting the multilingual functionalities of our methodology and investigating its suitability for other cognitive linguistic phenomena.

Keywords: Bidirectional Encoder Representations from Transformers; Long short-term memory networks; Natural language processing; Reverse Dictionary

1. Introduction

In this era of enriching computational linguistics, the need to upset the bridge between human language and machine interpretation become more imperative [1]. A reverse dictionary (RD) represents an impressive tool that addresses the user's descriptions by text to their meaning or idiom. Reverse dictionaries work in reverse to the traditional dictionary, which maps each word to its descriptive text. McNeill described this phenomenon in 1966. This resource is especially helpful for those who are temporarily unable to remember the precise expression, a condition known as the Tip-of-the-Tongue (TOT) phenomenon, but who may grasp the definition or notion of a word [2,3]. The user inputs the description; then, the RD returns a list of words that define the user's input, efficiently helping linguists, writers, new language learners, or anyone who wants to enrich his/her vocabulary. Relying on Siddique and Sufyan Beg in 2019 studies, RD is a crucial tool for filling the gaps in lexicons by directly addressing the issue of understanding what you want to say but not quite being able to put it into words [4]. Using the innovative natural language processing and Deep learning (DL) paradigms, a robust system has been crafted that not only finds the descriptors of the lexical items but is also used as a dictionary for linguistic search and learning a language [5,6].

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

Traditional information retrieval approaches have been used before, and they rely on matching words in the description to specify the suitable idiom. However, these approaches are limited in accuracy. In contrast, the deep learning approaches depend on context numerical features representing the semantic contextual feature set for the description. LSTM handles sentences as a sequence and extracts long-term features from sentences. While BLSTM takes feature sets from definition sentences from forward and backward, it provider a deeper understanding of semantics [7].

One aspect to consider in the proposed system novel is the dataset collected from the Web. The dataset covers large semantic fields and subjects. The scraped dataset is then pre-processed and cleaned to suit the contextual discriminative features extraction step. Modelling a robust system that can grab the features is the crucial phase. So, we worked on the study of the influence of Long Short-Term Memory networks (LSTMs) [8], Feedforward Neural Networks (FNN) [9], and Gated Recurrent Units (GRU) [10] on text feature extraction in this study. This study used cosine similarity and mean square error to compare the feature extraction models. This research contributed to the machine learning approach in computational linguistics in applying reverse dictionaries. These contributions include:

- Innovative RD Methodology: this research designs a unique approach to address the challenge of mapping word meanings to their appropriate contexts. Therefore, this work targeted the building and innovation of a pioneering paradigm that enables the translation of word meaning into concise, contextual definitions while enhancing the RD technique.
- Advanced DL Techniques: Our study leverages cutting-edge DL algorithms to dissect the intricacies of language, setting new benchmarks for accuracy and efficiency in linguistic processing.
- Framework for Future Research: The techniques and findings presented in our paper provide a solid foundation for future explorations in artificial intelligence, potentially leading to further breakthroughs in the domain.
- Collecting an innovative dataset handling the subject of e-learning from linguistic applications. This dataset used to train the model that address a set of idioms to its descriptive contexts.

The rest of this paper is organized as follows: Section 2 provides an overview of the works related to our research, offering insights into the previous studies that have shaped the current landscape of RD technologies and the treatment of the Tip-of-the-Tongue phenomenon. Section 3, Methodology, is divided into four subsections: Dataset details the data collection process; Data pre-processing describes the techniques used to refine the raw data for model input; Modelling outlines the design and structure of the neural network models employed; and Evaluation Measures discusses the metrics used to assess the performance of our models. In Section 4, Results, we present the outcomes of our experiments, providing a critical analysis of the model performances. Finally, Section 5 concludes the paper with a summary of our findings, contributions to the field, and potential avenues for future research.

2. Related Works

In this section, our attention is directed towards the prior research conducted in RD technology. We intend to examine various algorithms and methodologies employed in experimentation and delve into a selection of pre-existing artificial intelligence models constructed to address the complexities involved in executing RD tasks.

Paper [11] introduces ”WantWords,” an open-source online RD system that significantly improves upon the performance of existing English RD systems and uniquely offers support for Chinese and cross-lingual English-Chinese and Chinese-English RD queries. It is designed to be user-friendly, efficiently assisting users in overcoming the tip-of-the-tongue problem and aiding new language learners by quickly and easily finding the words, they need.

The study [7] presents an enhancement of the Neural RD (NRD), which maps word embedding from input definitions into a word embedding of the defined word using neural networks. The paper addresses the issue of insufficient accuracy in previous NRDs by employing novel combinations of neural networks, specifically a multi-layer fully connected network with bypass structures (CFNN) adjusted by LSTM output. The research found that the BiLSTM+CFNN model is comparable to, and in some aspects superior to, the commercial OneLook RD, especially when further enhanced with noising data augmentation. The study also explores the reasons behind the success of the BiLSTM+CFNN model, attributing it to the bypass structure of the CFNN and the balanced capacity of both LSTM and CFNN components.

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

The study [12] explores incorporating BERT into RD tasks, focusing on generating a target word from a given description. It proposes a novel method to adapt BERT, which typically relies on byte-pair-encoding (BPE) subword encoding, for this specific task. Furthermore, the paper addresses cross-lingual RD challenges by utilizing Multilingual BERT (mBERT), allowing efficient task execution using a single sub-word embedding without needing alignment between languages. Impressively, the study highlights that mBERT can deliver significant cross-lingual RD performance even without a parallel corpus, relying solely on monolingual data.

The paper [13] introduces a dual-way neural dictionary designed to retrieve words from definitions and simultaneously generate definitions for queried words. The model efficiently handles unknown words via embeddings by casting words and meanings into the same representation space through a shared layer and operates multi-task. The method demonstrates promising results on previous benchmarks without requiring additional resources. Moreover, the outputs of the model are preferred by human annotators in both reference-less and reference-based evaluations, indicating its practicality. The analysis within the study suggests that the incorporation of multiple objectives enhances the learning process.

The paper [14] introduces a novel open-source RD system to assist new language learners, anomia patients, and those facing tip-of-the-tongue problems. It focuses on Indian languages, for which RD support is currently unavailable. The system utilizes a transformer-based deep learning approach with the mT5 model. It employs the Translation Language Modelling (TLM) technique instead of BERT's Masked Language Modelling (MLM) to address existing limitations in reverse dictionaries.

The paper [15] focuses on the RD task, where a gloss is given as input, and the model is trained to generate a semantically matching word vector. The study evaluates a Transformer-based model with an added LSTM layer for this task in monolingual, multilingual, and cross-lingual zero-shot settings across five languages. The results show some improvement over the existing baseline in the CODWOE competition and explore the feasibility of cross-lingual approaches for the RD task, which is useful for solving tip-of-the-tongue problems and assisting new language learners.

The paper [16] addresses the lack of an RD resource for the Persian language and evaluates four different architectures for implementing a Persian RD (PREDICT). The models are assessed using (phrase word) pairs from online Persian dictionaries like Amid, Moein, and Dehkhoda, where the phrase describes the word. The best-performing model employs LSTM units with an additive attention mechanism and can suggest words effectively conveying the concept. Additionally, a new metric called "Synonym Accuracy" is introduced to evaluate the RD's performance, showing that the best model produces accurate results in at least 62% of cases within the top 100 suggestions, including synonyms of the target word.

The paper [17] introduces a multi-channel RD model designed to improve the handling of variable input queries and low-frequency target words. The model consists of a sentence encoder and multiple predictors, aiming to capture various characteristics of the target word from the input description. Evaluation of English and Chinese datasets, including dictionary definitions and human-written descriptions, demonstrates that the model achieves state-of-the-art performance, outperforming popular commercial RD systems on human-written descriptions. The paper also provides code and data for further exploration and application.

The paper [18] introduces the first KSAA-RD shared task, focused on developing an RD system for the Arabic language. The task involves two subtasks: Arabic RD and cross-lingual reverse dictionaries (CLRD). Participants compete to find the most similar word embeddings for given definitions (glosses) in Arabic or English. The winning team achieved rank metric scores of 24.20 for RD and 12.70 for CLRD. The paper provides insights into the methods used by participating teams and outlines the prospects for KSAA-RD.

In contrast to previously researched papers, our approach distinguishes itself by introducing an LSTM network alongside advanced text cleaning and lemmatization techniques and a novel extract word definitions function that harnesses NLTK's WordNet. This combination of techniques enhances the modelling phase, allowing us to capture the sequential nature of the text and extract contextually relevant definitions, thereby improving the overall performance of our RD system.

3. Proposed Approach

Our proposed methodology marks a significant progression in applying deep learning (DL) techniques for natural language processing (NLP), particularly in developing a Reverse Dictionary (RD). This paper outlines an integrated approach involving data collection, pre-processing, modelling, and evaluation metrics, emphasizing each stage's crucial role in creating our DL models. In the data collection phase, we capitalized on two distinct

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

datasets. The primary dataset was provided by the CODWOE 2021 competition, which laid a robust groundwork for our analysis. In addition to that, we scraped a rich dataset from various web platforms [19]. This dataset, gripped by many e-learning websites and web services, contributes to a rich and wide range of texts and schematics that relate to many subjects to make the model more generalized. The pre-processing operations target that data to convert the text into a feature set that can be handled by the machine learning models [20]. We conducted complicated text cleaning processes and lemmatizer [21] and also designed a function for extracting word embeddings; this function makes an important addition to computational linguistics by making it easier to get contextually relevant meanings from the vast vocabulary accessible through NLTK's WordNet [22]. Through the modelling step, we discussed many deep learning models, like Long Short-Term Memory (LSTM) networks, Feedforward Neural Networks (FNN), and Gated Recurrent Units (GRU). Each architecture consisted of many layers handling non-linearity features; the training process relied on cross entropy loss function, with the optimization function of Adam optimizer. By determining the difficulties associated with transient dependencies within the text, the LSTM was created to capture the sequential context of the text The model utilizes a series of sequential networks that convert the text into one set of features, which is later used by a classifier consisting of dense neural networks to predict the class of the input text. In the same principle, both FNN and GRU models were architected to improve the capability of LSTM by providing models to handle the complexities in the text. This multifaceted approach was able to systematically evaluate the advantages and disadvantages of each model within the context of RD technology. Our evaluation metrics, which included training loss, cosine similarity, and mean squared error, provided a thorough understanding of each model's performance. The combined achievements of these models, especially their low mean squared errors and high cosine similarity scores, outperform earlier benchmarks and show the potential of our methodology. Our work advances the state of RD technology by integrating multiple deep learning models. It creates a flexible framework that can be applied to different domains where the ability to comprehend and process natural language is crucial. Figure 1 presents the suggested methodology.

A. Datasets

This section provide a describe for the used dataset which are used in training the proposed approach.

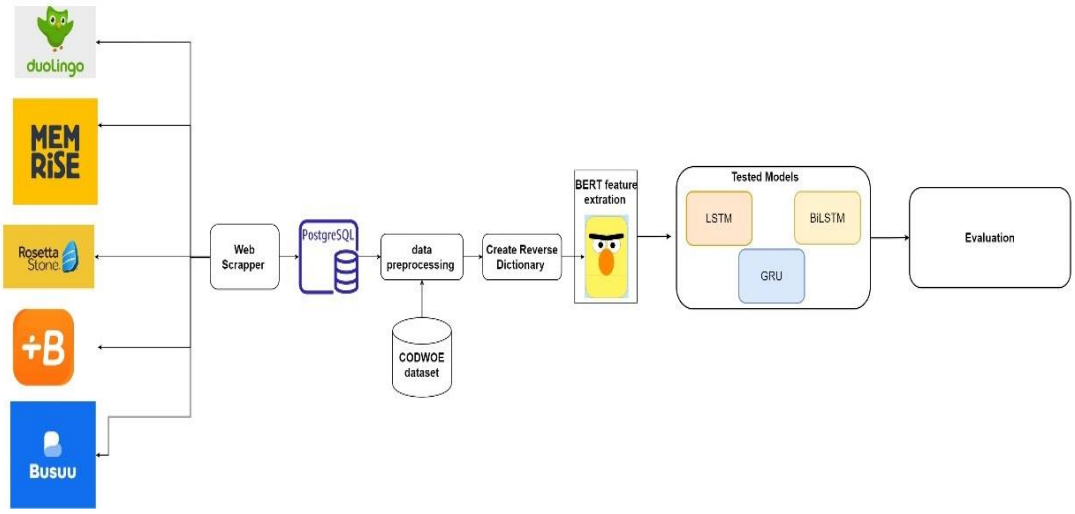

Figure 1. The proposed approach

- **CODWOE 2021 Dataset**

CODWOE 2021 has been utilized to focus on employing a thorough dataset, which was provided by the CODWOE 2021 competition [19]. This dataset is uniquely structured, encompassing glosses across five languages: English (en), Spanish (es), French (fr), Italian (it), and Russian (ru), with each gloss being represented through three distinct word embedding models. For our experiments, we focused on the English language component of the dataset. This subset is organized into three sets: a training set with 43,608 samples, a development set comprising 6,375 samples, and a test set containing 6,208 samples. Despite the uniform distribution of sample sizes across the languages, the dataset intricately showcases the linguistic phenomena of

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

polysemy, where a single word may have multiple glosses, and synonymy, where a gloss could correspond to more than one word. Notably, the dataset is training and development phases format obscures the direct correspondence between words and their glosses, instead providing various embedding for each gloss. This design choice introduces a challenging yet fascinating aspect to our work, as it necessitates the development of a nuanced understanding of gloss-embedding relationships without relying on explicit word gloss mappings. The dataset employed in this research, as illustrated in Figure. 2, showcases a snippet of the English development set from the CODWOE 2021 competition, featuring glosses alongside their corresponding Sngs, char, and Electra embedding, which forms the basis for our transformer-enhanced reverse dictionary model. These embeddings are extracted from pre-training approaches to get semantics feature values for each definition sentence. All embedding generators convert the descriptive phrase into dense vectors that reflect the semantic meaning of the phrase [23].

- **Scrapped Dataset**

The data collection step for the research paper is initiated by a thorough data acquisition process that employs scraping techniques. Scraping is a method used to automatically harvest information from the internet, where scripts or programs navigate through web pages, identify relevant data, and then gather and record it for further use. This technique is particularly beneficial in gathering large amounts of data quickly and efficiently, which is vital in training robust DL models.

Using the scrapping technique, we gathered various datasets from two primary sources. Two sources were scrapped. First, there are e-learning programs such as Duolingo, Rosetta Stone, Memrise, Babbel, and Busuu, with reviews relating to the users' opinions of these applications [24]. The scrap gathered about 5000 reviews from each application. This method yielded a rich dataset of reviews that was enough to train a complete language model as it represented a wide range of subjects related to e-learning languages. In addition, the scrapping system is used to collect reviews from Uber Eats, a widely used food delivery service. So, we ensure that our model can work under various subjects with a wide range of vocabulary and cover definitions that reflect the idioms used in daily speech.

- **Scrapped Dataset**

The data collection step for the research paper is initiated by a thorough data acquisition process that employs scraping techniques. Scraping is a method used to automatically harvest information from the internet, where scripts or programs navigate through web pages, identify relevant data, and then gather and record it for further use. This technique is particularly beneficial in gathering large amounts of data quickly and efficiently, which is vital in training robust DL models.

Using the scrapping technique, we gathered various datasets from two primary sources. Two sources were scrapped. First, there are e-learning programs such as Duolingo, Rosetta Stone, Memrise, Babbel, and Busuu, with reviews relating to the users' opinions of these applications [24]. The scrap gathered about 5000 reviews from each application. This method yielded a rich dataset of reviews that was enough to train a complete language model as it represented a wide range of subjects related to e-learning languages. In addition, the scrapping system is used to collect reviews from Uber Eats, a widely used food delivery service. Therefore, we ensure that our model can work under various subjects with a wide range of vocabulary and cover definitions that reflect the idioms used in daily speech.

	id	gloss	sgns	char	electra
0	en.dev.1	A meal co	[0.923365:	[-0.121425	[0.3655579984, -0.1910238415, 0.0170905143, 0
1	en.dev.2	Located al	[-0.086880	[0.317635(	[-0.6527376175, -0.6509178877, 0.1359305233, 0...
2	en.dev.3	to do som	[1.640457(	[-0.174104	[-0.4694166481, -0.1934900135, 0.1299305409, -...
3	en.dev.4	The clitori	[0.694295:	[0.295901(	[-1.3414018154, -0.2901522815, -0.022413671, -...
4	en.dev.5	Excellent	[0.790026(	[-0.234957	-1.6138648987, -1.0392408371, -0.1034375206,
...	...				
6370	en.dev.63	To promo	[1.682262:	[1.501108(	[-1.1007618904, -0.4101574421, 0.2163364291, -...
6371	en.dev.63	Certificate	[0.365935:	[0.627070(	[0.2797732949, -0.176243335, 0.3984850347, 0.0.
6372	en.dev.63	Relating t	[0.052371:	[-0.075137	[-1.7170902491, -0.2992493808, 0.3077795506, 0.
6373	en.dev.63	An act of t	[2.949348(	[-0.247341	[-0.8273163438, -0.1927744001, 0.0762535185, 0...
6374	en.dev.63	Of or relat	[-0.040898	[0.074327(	[-0.59497118, -0.2254087329, 0.2311077416, -0

Figure 2. A snippet of the English development set from the dataset

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

B. Data Pre-processing

In this study, the text datasets were pre-processed using various techniques to refine and clean the textual data. These pre-processing steps refine the input text to convert it to a new standardized form. One of the crucial was a clean function that changed all text to the low case by deleting the special characters and adding sentence-ending predios to ensure consistent sentences in the dataset. paCy's library functions are used in tokenization, lemmatization of the text, and deletion of punctuation marks, as well as stopping words, numbers, special characters, and symbols. This function keeps just the important words in the context representing the information processed. The essential part of the proposed methods is a function known as "extract word definitions," which means the reverse dictionary (RD) base and is viewed as advanced in computational linguistics. The annotation process was finished with the WORDNET lexicon from NLTK tools, which adds the definition of the words in the text. Labelling can reveal all text words with their context semantic meaning as an essential feature. This process is accomplished by tokenizing the text and lemmatize the words. Then, the function interacts with the WORDNET dictionary to return the lamma of each word with meaning. This connection goes beyond basic annotation and provides detailed information on how the word is used in multiple settings. this contextual retrieval from text is not just a definition specifying; it is a clever interpretation of the word to determine the meaning of the word in the real-world text. The proposed paradigm specifies the word meaning, enriching the lexicon with rich semantic meanings crucial for the RD approach, and addresses the word with the closest definition. As a result, the created RD provides the users with an accurate and relevant definition, which reverses its work with traditional word-to-definition addressing. Semantic embedding into the dataset will be a powerful tool for specifying the definition with complicated titles for exploring the language. In Figure 3, a part of the dataset, where each entry represents just one word with its contextual features sector, provides a prosperous semantic relationship with the proposed RD system. To focus on this process, this proposed paradigm uses a function called “extract word definition,” which is the center point of this work because it labels the words coming within the text by exploiting the linguistic corpora using the NLTK’s WORDNET.

	word	definition
0	blemish	a mark or flaw that spoils the appearance of s...
1	vocal	music intended to be performed by one or more ...
2	genre	a kind of literary or artistic work
3	hindustani	a native or inhabitant of Hindustan or India
4	classical	traditional genre of music conforming to an es...
5	music	an artistic form of auditory communication inc...
6	illegal	prohibited by law or by official or accepted r...
7	sale	a particular instance of selling
8	especially	to a distinctly greater extent or degree than ...
9	drug	a substance that is used as a medicine or narc...

Figure 3. Reverse dictionary dataset

Subsequently, attention was directed towards systematically generating word definition pairs from the processed and clarified text. The pairs were systematically recorded in a new column and then carefully processed to remove any missing values and duplicate entries, guaranteeing the dataset's originality and clarity.

An important advancement in the pre-processing stage was the incorporation of BERT, an advanced, pre-trained transformer model, for producing word embeddings. Using the create embeddings function; each word definition was converted into a compact numerical representation, capturing the intricate contextual subtleties and semantic depth of the language employed in the reviews. These embeddings are crucial for the model to be trained with suitable tokens of values. In the end, the text data is transformed and split into embeddings. From the embedding perspective, each word is addressed as a numerical label representing the semantic meaning in the context of speech. Splitting data into training and testing sets is essential in any machine or deep learning operation as it conducts model crafting and evaluation. In this instance, the dataset of embeddings array and label’s arrays were split into 70% training set and 30% for testing. The test set is divided into validation and testing parts, each consisting of 15% of the initial data in general. Consequently, the data distribution resulted in 70% of the data for training the model, 15% for validating and fine-tuning during the iterative process of model development, and the final 15% for testing, which provides an unbiased evaluation of the model’s performance on unseen data.

These meticulous pre-processing steps transformed the raw, unstructured text into a clean, structured, and semantically rich dataset, laying a solid foundation for the subsequent modelling and analysis stages in our pioneering research on RD technology.

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

In Figure 4, the table presents word embeddings generated using BERT, arranged in a tabular layout. Each word from the dataset, ranging from "blemish" to" Oxford," is accompanied by its respective definition and transformed into a high-dimensional vector representation. These embeddings, comprising arrays of numerical values, encapsulate the intricate semantic characteristics of each word's definition. Notably, the embeddings facilitate nuanced language comprehension and processing by machine learning models, enabling them to more effectively grasp the subtle nuances of word meanings and contexts.

	word	definition	embeddings
0	blemish	a mark or flaw that spoils the appearance of S..	[-0.9068658, -0.5096633, -0.9883213, 0.9314614
1	vocal	music intended to be performed by one or more	[-0.9223473, -0.5607347, -0.9899275, 0.9285204
2	genre	a kind of literary or artistic work	[-0.897419, -0.40232563. -0.9355683, 0.8578806
3	hindustan	a native or inhabitant of Hindustan or India	[-0.9639249, -0.5413992, -0.91502523 0.927688
4	classical	traditional genre of music conforming to an es.,	[-0.9315985, -0.44987753, -0.97404635, 0.91803
...			
17132	sheen	the visual property of something that shines W.	[-0.9042184, -0.22620241, -0.68816197, 0.83329
17133	indignatic	a feeling of righteous anger	[-0.77694875, -0.14566922 0.17857423, 0.43306
17134	outrage	a feeling of righteous anger	[-0.77694875, -0.14566922 0.17857423, 0.43306
17135	bribery	the practice of offering something (usually mo	[-0.8089142, -0.288371, -0.516584, 0.6310152,
17136	oxford	a city in southern England to the northwest of	-0.89198667 -0.44160625 -0.6400832, 0.79143

Figure 4. Word embedding

C. Modelling

In the modelling phase of the proposed approach, detailed in our research paper, we embarked on a comprehensive comparative analysis between three prominent architectures FNN, GRU, and LSTMs.

1. Feedforward Neural Network (FNN) is a fundamental type of artificial neural network used in deep learning and deep learning. It belongs to the category of feedforward networks because the data flows in one direction, from the input layer through one or more hidden layers to the output layer, without any loops or feedback connections. A variety of tasks, including regression, classification, and pattern recognition, have employed FNNs. Specifically, the FNN model is designed to classify words corresponding to their definitions through BERT embeddings. The architecture of this model follows a usual feedforward design that has been customized to meet the specific needs of the work. The FNN model's input layer accepts BERT embeddings as input, which are 768-dimensional vectors that reflect contextual information derived from word definitions as in Figure 4.The embeddings undergo processing via a sequence of fully connected layers. The first hidden layer consists of 512 neurons and is followed by a Rectified Linear Unit (ReLU) activation function that introduces non-linearity into the model. Dropout layers with a dropout rate 0.5 are applied after each ReLU activation to prevent overfitting. following the first hidden layer is a second one with 256 neurons and another ReLU activation, followed by dropout.

Finally, the output layer of the FNN model consists of a fully connected layer with the number of neurons equal to the number of unique words in the dataset, which is the classification target. This output layer does not have an activation function since it's used for multi-class classification, and the model aims to provide raw scores for each class. The model uses the cross-entropy loss function during training to compute the error between its predictions and the ground truth labels. The Adam optimizer modifies the model's weights and minimizes the loss. To help prevent overfitting, early stopping is used as a regularization technique to monitor the validation loss and stop training if the loss does not improve for a predetermined number of epochs.

2. A particular kind of recurrent neural network (RNN) architecture called the Gated Recurrent Unit (GRU) was designed to solve the vanishing gradient issue, which can be difficult to train conventional RNNs. Like the Long Short-Term Memory (LSTM) network. GRUs are utilized for sequential data processing tasks like speech recognition, time series analysis, and natural language processing, as shown in Figure 6, the GRU layer and the fully connected

(FC) output layer make up the two primary parts of the GRU model's architecture. The GRU layer handles the sequential input data processing and feature extraction that comes with it. The

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

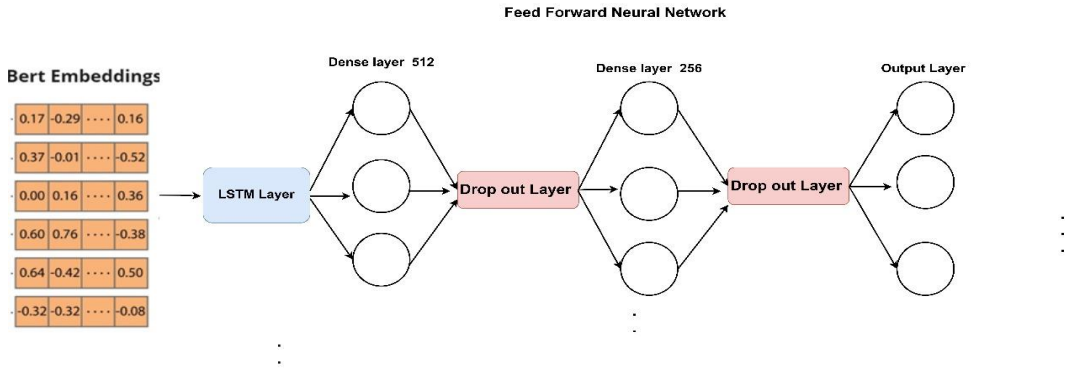

Figure 5. FNN architecture

GRU model is tuned by batch size, hidden neurons, and input size. Throughout the input procedure, the data is input to the GRU input layer, which feeds the batch first, enabling effective parallelism. The GRU internal layer processes the text and introduces a set of hidden values. These hidden values represent semantic and temporal values that define the text pattern for a specific context. The model converts the text into a feature from the last layer in the hidden neurons to obtain a single feature vector. The input text is converted from a sequence of words into a fixed-size representative array of numerical values and then entered into a fully connected layer (FC). This FC has the number of neurons as several output classes; its functionality is to classify the text. The retrieved features are used as an input to generate the output from that model. The probability of each class represents the model's prediction process output. These models work well for various NLP applications.

3. Extended Short-Term Memory Networks (LSTMs) are one of the neural network architectures. The design of LSTM targeted a specific objective in the text, which is handling the text's sequential nature by catching the pattern of the sequential data. This paper has developed LSTM to process text by gathering the temporal pattern that is considered a feature that is significant for the structure of any text or language. The LSTM was executed by using PyTorch. The module is a library for designing deep neural networks and complex LSTM models. The proposed model features several neurons in the input layer, a hidden layer with its number of neurons, and a layer equal to the number of output classes. These parameters can be tuned to tackle various degrees of complexity in the linguistic task.The LSTM is the vital element in the proposed model, which is in charge of processing the sequential text by using some internal processes to catch the long-term patterns of the text sentence. The batch size is indicated from the beginning on the first input layer of the model. Subsequently, the fully connected layer links the hidden layer to reach the output layer, corresponding finally to the number of words in the sentence. In the forward pass, we initialize the hidden and with a clean slate for each new batch. The LSTM processes the input sequence, accompanied by these initial states, yielding the final output through the hidden state of the last time step, which is then passed through the output layer to produce the predictions.

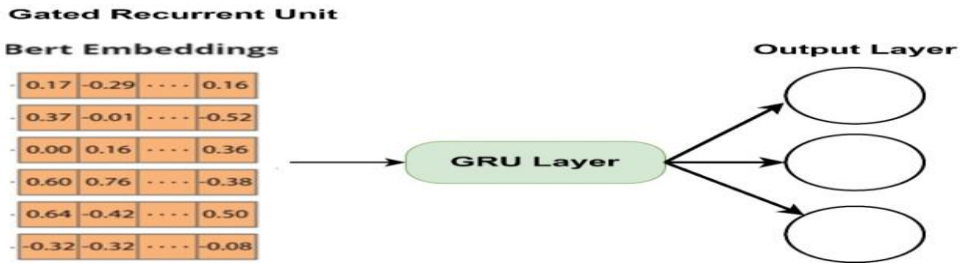

Figure 6. GRU architecture

The LSTM model's parameters are initialized to accommodate the size of the BERT embeddings used as input and to capture the complexity of the dataset with an appropriate number of hidden units and layers. We employ a batch size of 64, balancing the computational efficiency and the model's ability to generalize across the dataset. Training this model involves a loop with early stopping, a strategy that monitors the validation loss and halts the training if there's no improvement, effectively preventing overfitting. We employ the Adam optimizer for its adaptive learning rate, which helps us converge to optimal solutions more efficiently, Figure 7 illustrated the LSTM architecture.

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

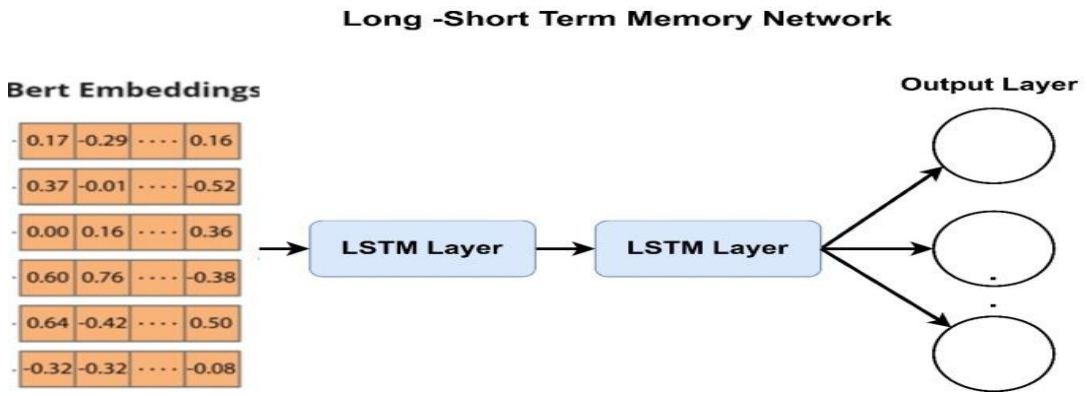

Figure 7. LSTM architecture

D. Evaluation Measures

In this research, the evaluation of the proposed models was multifaceted, incorporating a variety of metrics to assess performance comprehensively. Each measure provides a different perspective on the model's ability to understand and generate language. Three primary evaluation measures were used: Training Loss, Cosine Similarity, and Mean Squared Error (MSE).

1. Training Loss: measures how well the model fits the training data. Lower values indicate a better fit. It is calculated during the training process at each epoch. For a classification task using cross-entropy loss, the training loss L for a single observation is given by Eq.(1):

$$L(y, y^{-}) = \sum_{i=1}^{n} yi \,.\log\left(\frac{yi}{y^{-}}\right) \qquad (1)$$

Where L is the number of classes, y is the binary indicator (0 or 1) if class label is the correct classification for observation o, and y^{-} is the predicted probability observation o is of a specific class.

2. Cosine Similarity: is a metric used to measure how similar the word embeddings are to each other, which can indicate semantic similarity.as in Eq.(2). It is particularly useful when comparing the angle between two vectors in a multi-dimensional space. The Cosine Similarity S between two vectors A and B is defined as:

$$S(A, B) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \qquad (2)$$

Where Ai and Bi are components of vectors A and B, respectively.

3. Mean Squared Error (MSE): measures the average squared difference between the estimated and actual values. In word embeddings, it can reflect how closely the predicted embedding matches the actual embedding. The MSE for n predictions is calculated as follows in Eq.(3)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - Y_i\hat{}\right)^2 \qquad (3)$$

where Yi is the true value of the ith data point, and Yˆi is the corresponding model's prediction.

Each of these measures offers a lens through which the performance of our models can be quantitatively assessed. Training loss allows us to monitor the fitting process, Cosine Similarity gives insight into the semantic accuracy of the embeddings, and MSE provides a clear numerical indicator of the model's prediction accuracy. Together, they form a robust framework for evaluating the proposed models' efficacy in the RD generation task.

4. Results

In the following two sections, we illustrate the results and the findings of the training and testing the proposed model for both CODWOE, and the scrapped datasets.

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

A. Results On The Codwoe Dataset

Figure 8, shows three plots for training loss across epochs for three different neural network architectures: LSTM, FNN, and GRU.

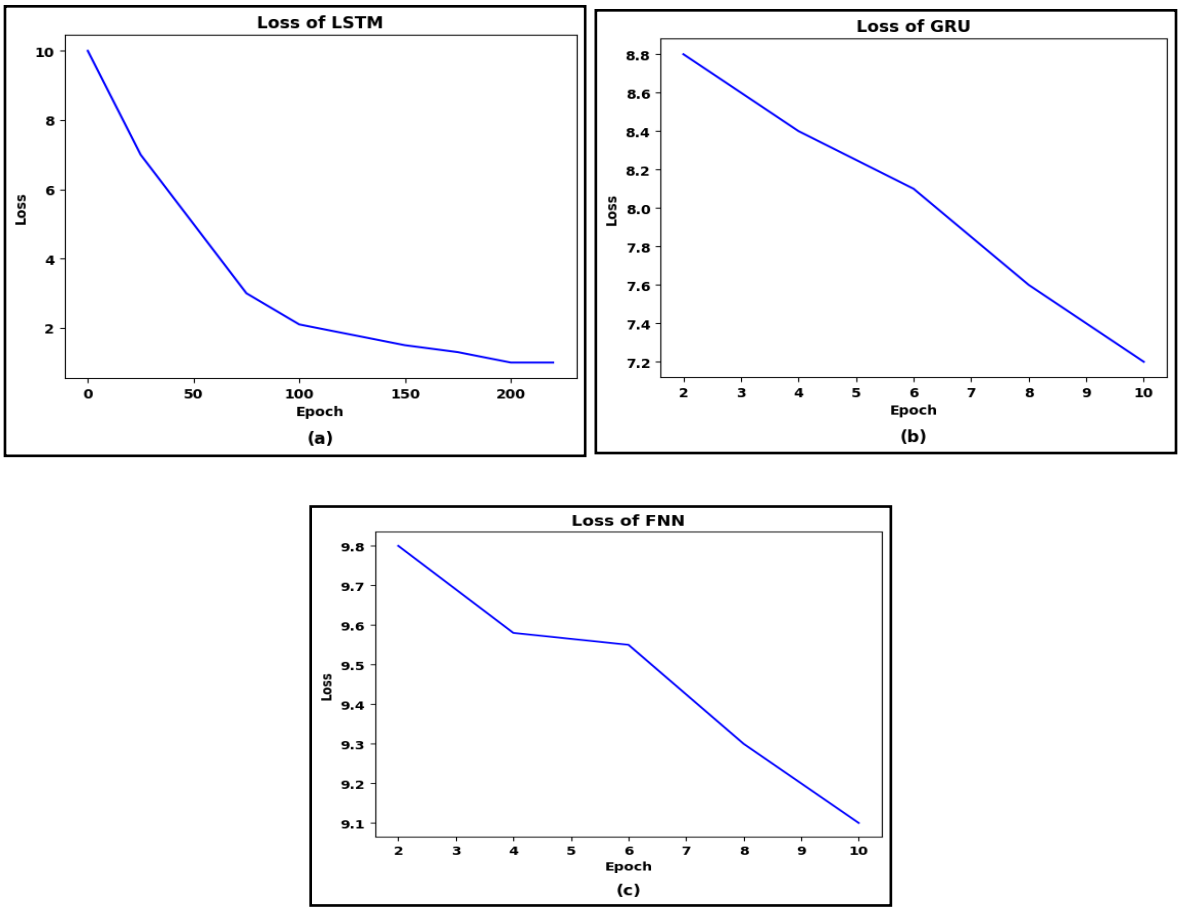

Figure 8. Training loss of the models on CODWOE dataset

The training loss results for the LSTM model in Figure 8-a illustrate a pronounced decline, commencing from an initial loss of approximately 10 and plummeting to a level close to 0, as indicated by the plot labeled" Loss of LSTM." The steep decline in loss during the initial epochs signals substantial learning as the model assimilates the dataset's structure. The curve flattens as epoch's increase, suggesting the model's convergence and diminishing returns in learning from additional training.

For the FNN model Figure 8-b, the" Loss of FNN" training loss graph depicts a moderate but consistent decrease across the epochs presented, starting just below 10 and approaching a loss value of around 9. This decline suggests that the FNN is learning, but without knowing the behaviour beyond the 10th epoch, it is unclear if this model will continue to improve or begin to over fit. In the GRU model's case in Figure 8-c, the "Loss of GRU" plot shows a relatively linear reduction in loss from just below 9 to just above 7 across the 10 epochs depicted. This steady decrease indicates the GRU model is learning from the training data, yet the absence of a flattening curve could imply that the

B. Results On Web Scrapped Dataset

The provided results from the training processes of LSTM, FNN, and GRU models presented in Figure 9 reveal distinctive patterns of learning and convergence.

The LSTM model is training in Figure 9-a began with a loss of 8.4826. Over 200 epochs, it demonstrated a consistent decrease in loss, reaching a point of early stopping with a significantly reduced loss of 0.2362. This trajectory, visible in the "Loss of LSTM" plot, signifies a robust learning process where the model effectively captures the temporal dependencies in the dataset. The LSTM layers' ability to learn complex patterns is shown by

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

their gradual convergence to a lower loss value. This is important for tasks involving sequences, like time-series forecasting or natural language processing. The regularization technique of early stopping at epoch 200 suggests that the model's performance on the validation set stopped improving, which led to the training being stopped to avoid overfitting and unnecessary computational expense. The GRU model's training, as in Figure 9-b, also displays a pattern suggestive of overfitting. Starting with a training loss of 8.8 and ending with 7.2198, the model is training loss steadily dropped consistently across 18 epochs. However, the validation loss went up from 8.631 to 8.821 again, a clear sign that the model's generalizability is failing. This is reflected in the "Loss of GRU" plot, showing the declining training loss curve. Early stopping was again employed as a preventive measure against overfitting, interrupting the training process when the validation loss failed to improve. In contrast, the FNN model in Figure 9-c displays a training pattern with initial promise but quickly raises concerns. Although the training loss decreased from 8.7120 to 7.9013 within the first 11 epochs, the validation loss increased from 8.6312 to 9.0, indicating a divergence between the model's performance on the training data and its generalization to new data. Such a discrepancy is symptomatic of overfitting. The increase in validation loss alongside the decrease in training loss, as observed in the "Loss of FNN" plot, suggests that while the model was becoming better at fitting the training data, it was concurrently losing its ability to generalize, thus leading to early stopping to avert further over fitting.

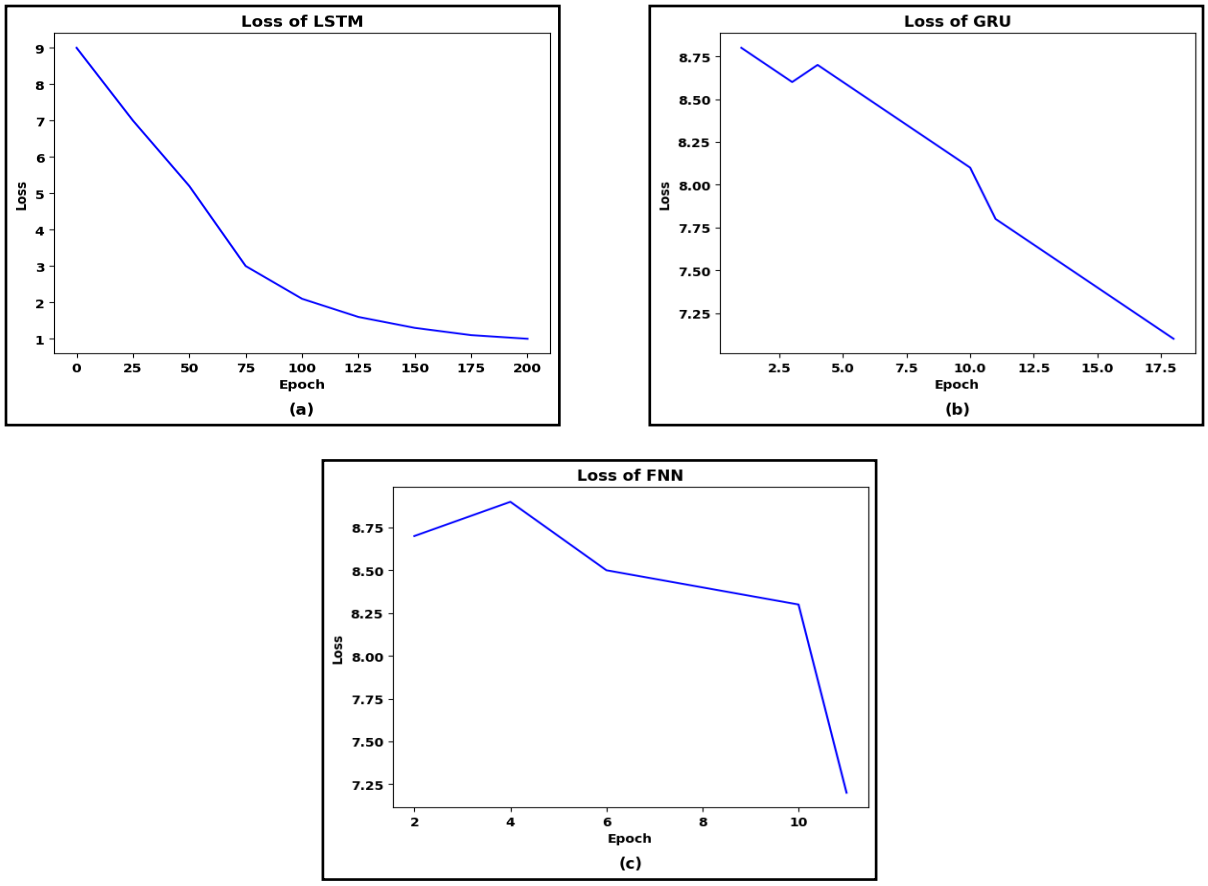

Figure 9. Training loss of the models on scrapped dataset

The results refer to the necessity for accurately noticing the training and validation losses to check the model efficiency. They also put the shade on the critical role of the early stooping technique as a mechanism to protect the model from overfitting, in addition to the need for hyper parameter tuning, regularization, or maybe another architecture for the model to add generalization to the FNN and GRU models.

C. Discussion

Table 1 illustrates the differences between various DL modes, focusing on their performance by comparing them on two datasets: benchmarked CODWOE and scarped dataset. The metrics adopted for evaluating the models are the cousin similarity and mean squared error (MSE), essential in addressing any model's accuracy and error rate in precision tasks. By using the CODWOE dataset, three models were conducted: first, a feedforward neural

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

network; second, a gated recurrent unit (GRU); and last, an LSTM network. The LSTM model showed the best performance, scoring a Cosine Similarity score of 0.9836. Additionally, LSTM showed a noticeably low MSR of 0.02005, scoring a minimal score from the expected value. In contrast, the GRU model registered a low cosine similarity of 0.0749, with an MSE of 0.5013, which indicates inaccuracy in the prediction process. The FNN model demonstrated a moderate performance with a Cosine Similarity equal to that of the LSTM model but with a significantly higher MSE of 21.93, suggesting consistent outputs but with a considerable error rate. interestingly, the performance trends were consistent when applied to the Scrapped Dataset. The FNN and LSTM models scored Cosine Similarity scores of 0.8541 and 0.9520, respectively, with the LSTM maintaining its superior performance with the lowest MSE of 0.04006. The GRU model's performance appeared reversed in this dataset, showing a Cosine Similarity of 0.5013 and an MSE of 0.07490, indicating a better alignment with the target yet a higher error rate than its performance on the CODWOE dataset.

Table 1: Comparatives performance of DL-models

Dataset	Model	Cosine Similarity	Mean Squared Error
CODWOE	FNN GRU LSTM	0.9097 0.0749 0.9836	21.93 0.5013 0.02005
Scrapped Dataset	FNN GRU LSTM	0.8541 0.5013 0.9520	33.77 0.07490 0.04006

This study displays a substantial leap in performance over prior research conducted on the same dataset, as depicted in Table 2. The results presented in the comparative performance table on the CODWOE dataset demonstrate a significant advancement in the performance of LSTM models as developed in our work. With a mean squared error (MSE) of just 0.02005 and a cosine similarity score of 0.9836, our LSTM model outperforms prior efforts cited as references. For instance, the model mentioned in [15] demonstrates a range of outcomes with their best LSTM, achieving an MSE of 0.913 and a cosine similarity of 0.156, substantially lower than our model's performance. They also experimented with a Bidirectional LSTM (BiLSTM), a combined approach, and a Multilingual LSTM, all of which yielded less favorable results than our model. The BiLSTM-LSTM model from [25] closely competes with an MSE of 0.895 and a cosine. The similarity of 0.153 still falls short of the standards set by our LSTM model. A hybrid model referenced in [26] also significantly lags with an MSE of 2.0157 and a cosine similarity [15] of 0.4029.

Table 2: Comparison of LSTM model performance.

Ref	Model	Cosine Similarity	MSE
[15]	LSTM BiLSTM	0.913 0.938	0.156 0.125
	Combined Multilingual LSTM	0.909 1.184	0.139 0.003
[26]	Hybrid Model	2.0157	0.4029
[25]	BiLSTM-LSTM	0.895	0.153
Our Model	LSTM	0.02005	0.9836

5. Conclusion

The reverse dictionary is a flip for the concept of the regular dictionary. In RD, the user describes a concept, and the RD finds the word or the idiom that defines the provided description. RD can help people learn language by increasing their vocabulary and solving the Tip-of-the-tongue moment issue. Sometimes, someone knows the description of something, but the word escapes him, so here, the RD solves this problem. This research examines three DL approaches (LSTM, BiLSTM, and GRU) with BERT embeddings extracted from phrases in the dataset. Our tests were conducted using two datasets; one was a scrapped dataset, and the second was a benchmarked dataset. By harnessing the capabilities of LSTM networks, we have demonstrated that machines can mirror the complex cognitive processes of human language retrieval. The comparative analysis has revealed that the proposed

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

model that relies on BERT semantic embeddings with LSTM can generate high cosine similarity scores, reflect accurate semantic representation, and maintain low mean squared error rates, underscoring their predictive precision. The suggested approach could be used as a guide to create a robust RD that exceeds existing paradigms that depend on IR or other deep learning architectures. The efficacy of our model sets a new standard for RD technology, rendering it an invaluable tool for linguists, writers, and language learners alike. Furthermore, our findings have implications for various applications, such as helping creative writing and supporting the communicative strategies of non-native speakers. Given their potential for multilingual support and integration with media types other than text, reverse dictionaries seem to have a bright future. We believe that further research will expand on our findings through looking into the incorporation of more sophisticated NLP methods and extending the use of reverse dictionaries. Our research shows how deep learning can close the gap between human cognition and computational efficiency, and we hope that this will lead to more advancement in this rapidly advancing field.

Conflicts of Interest

We would like to declare that there is no conflict of interest in this research

Author Contributions

The contribution of the researcher's team is as follows; DR. Ahmed Bahaaulddin is working for conceptualization and methodology. While DR. Sian Sabeeh is working on software and writing original draft preparation. Mr. Ali Sami helping is both writing the original draft and editing. Last Dr. Aliaa Alkhraz helped in writing, review and editing.

References

[1] Khishigsuren T, Bella G, Batsuren K, Freihat AA, Nair NC, Ganbold A, et al. Using linguistic typology to enrich multilingual lexicons: the case of lexical gaps in kinship. ArXiv Prepr ArXiv220405049 2022.
[2] Brown AS. A review of the tip-of-the-tongue experience. Psychol Bull 1991;109:204–23. https://doi.org/10.1037//0033-2909.109.2.204.
[3] Brown R, McNeill D. The "tip of the tongue" phenomenon. J Verbal Learning Verbal Behav 1966;5:325–37. https://doi.org/10.1016/s0022-5371(66)80040-3.
[4] Siddique B, Sufyan Beg MM. A Review of Reverse Dictionary: Finding Words from Concept Description. Commun Comput Inf Sci 2019:128–39. https://doi.org/10.1007/978-981-15-1718-1_11.
[5] Ive J. *Natural Language Processing: A Machine Learning Perspective* by Yue Zhang and Zhiyang Teng. Comput Linguist 2022;48:233–5. https://doi.org/10.1162/coli_r_00423.
[6] Jiang K, Lu X. Natural Language Processing and Its Applications in Machine Translation: A Diachronic Review. 2020 IEEE 3rd Int Conf Safe Prod Informatiz 2020. https://doi.org/10.1109/iicspi51290.2020.9332458.
[7] Morinaga Y, Yamaguchi K. Improvement of Neural Reverse Dictionary by Using Cascade Forward Neural Network. J Inf Process 2020;28:715–23. https://doi.org/10.2197/ipsjjip.28.715.
[8] Lindemann B, Müller T, Vietz H, Jazdi N, Weyrich M. A survey on long short-term memory networks for time series prediction. Procedia CIRP 2021;99:650–5. https://doi.org/10.1016/j.procir.2021.03.088.
[9] Sazlı MH. A brief review of feed-forward neural networks. Commun Fac Sci Univ Ankara Ser A2-A3 Phys Sci Eng 2006;50.
[10] Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. ArXiv Prepr ArXiv14123555 2014.
[11] Qi F, Zhang L, Yang Y, Liu Z, Sun M. WantWords: An Open-source Online Reverse Dictionary System. Proc 2020 Conf Empir Methods Nat Lang Process Syst Demonstr 2020. https://doi.org/10.18653/v1/2020.emnlp-demos.23.
[12] Yan H, Li X, Qiu X, Deng B. BERT for Monolingual and Cross-Lingual Reverse Dictionary. Find Assoc Comput Linguist EMNLP 2020 2020. https://doi.org/10.18653/v1/2020.findings-emnlp.388.
[13] Chen P, Zhao Z. A unified model for reverse dictionary and definition modelling. ArXiv Prepr ArXiv220504602 2022.
[14] Mane SB, Patil HN, Madaswar KB, Sadavarte PN. WordAlchemy: A Transformer-based Reverse Dictionary. 2022 2nd Int Conf Intell Technol 2022. https://doi.org/10.1109/conit55038.2022.9848383.
[15] Tran THH, Martinc M, Purver M, Pollak S. JSI at SemEval-2022 Task 1: CODWOE - Reverse Dictionary: Monolingual and cross-lingual approaches. Proc 16th Int Work Semant Eval 2022. https://doi.org/10.18653/v1/2022.semeval-1.12.
[16] Malekzadeh A, Gheibi A, Mohades A. PREDICT: persian reverse dictionary. ArXiv Prepr ArXiv210500309 2021.

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024

[17] Zhang L, Qi F, Liu Z, Wang Y, Liu Q, Sun M. Multi-Channel Reverse Dictionary Model. Proc AAAI Conf Artif Intell 2020;34:312–9. https://doi.org/10.1609/aaai.v34i01.5365.
[18] Al-Matham R, Alshammari W, AlOsaimy A, Alhumoud S, Wazrah A, Altamimi A, et al. KSAA-RD Shared Task: Arabic Reverse Dictionary. Proc Arab 2023 2023. https://doi.org/10.18653/v1/2023.arabicnlp-1.39.
[19] Pavao A, Guyon I, Letournel A-C, Tran D-T, Baro X, Escalante HJ, et al. Codalab competitions: An open source platform to organize scientific challenges. J Mach Learn Res 2023;24:1–6.
[20] Shanmugam R. Practical text analytics: maximizing the value of text data. J Stat Comput Simul 2019;90:1346. https://doi.org/10.1080/00949655.2019.1628899.
[21] S. M. Khan and A. R. Alzubaidi, "Transformer-Based Models for Natural Language Processing: A Review," *Journal of Computer and Communications*, vol. 10, no. 5, pp. 1-12, May 2022. doi: 10.4236/jcc.2022.105001.
[22] Millstein F. Natural language processing with python: natural language processing using NLTK. Frank Millstein; 2020.
[23] Sibaee S, Ahmad S, Khurfan I, Sabeeh V, Bahaaulddin A, Belhaj H, et al. Qamosy at Arabic Reverse Dictionary shared task: Semi Decoder Architecture for Reverse Dictionary with SBERT Encoder. Proc Arab 2023 2023. https://doi.org/10.18653/v1/2023.arabicnlp-1.41.
[24] T. J. Lee and M. H. Kim, "Enhancing Dictionary Learning through Transformer Architectures for Semantic Retrieval," International Journal of Computer Applications, vol. 175, no. 3, pp. 25-32, Nov. 2021. doi: 10.5120/ijca2021916790.
[25] Bendahman N, Breton J, Nicolaieff L, Billami MB, Bortolaso C, Miloudi Y. BL.Research at SemEval-2022 Task 1: Deep networks for Reverse Dictionary using embeddings and LSTM autoencoders. Proc 16th Int Work Semant Eval 2022. https://doi.org/10.18653/v1/2022.semeval-1.11.
[26] Ardoiz A, Ortega-Martín M, García-Sierra Ó, Álvarez J, Arranz I, Alonso A. MMG at SemEval-2022 Task 1: A Reverse Dictionary approach based on a review of the dataset from a lexicographic perspective. Proc 16th Int Work Semant Eval 2022. https://doi.org/10.18653/v1/2022.semeval-1.7.

DOI: https://doi.org/10.54216/FPA.170101
Received: October 28, 2023 Revised: February 25, 2024 Accepted: June 25, 2024