WORD PREDICTION

18BCE2401 – Binamra Neupane
19BCE2609 - Rishi Srikaanth
20BCE2075 - Ananya Redhu
20BCE2099 - Vyom Gupta
20BCE2857 - Garv Mittal

Abstract:
Word prediction is certainly considered one among the most basic chore for language fashions withinside the subject of NLP, which makes use of a language model to come across the joint distribution of natural language phrase sequences. We came across different algorithms and techniques used for predicting words. It also gave us an opportunity to learn how word prediction works. Out of all languages English language is easy to predict because it easier to segment. The link between the parts of a sentence in natural language, words, sentences, an d clauses are generally arranged in a tree-like fashion, syntactic structure is created by the reliance between separate parts in a sentence. In digitalized conversation emoticons are also important. It conveys the feeling without even saying a word. Emoji predictions are made using recurrent neural networks. On reviewing the research articles, researchers have demonstrated their techniques in a very pleasing way. Some papers have developed better algorithms, surpassing the previous algorithms. Algorithms can be used on any kind of datasets. However, better the quality datasets greater the performance. Existing models for the bangla word prediction is not alluring. To forecast the likely terms, the Stupid BackOff algorithm (SBO) is used and it works superior for lager datasets. SBO uses a backoff score that may be written as Lambda and equal instead of discounting since it uses proper frequencies instead of discounting. Some languages do not have proper dataset hence, difficult to predict the word. For instance, urdu the language has remained under resourced due to a scarcity of corpora and datasets for conducting numerous computational tasks. Word embeddings have gotten plethora of interest in contemporary years because their intrinsic usefulness in Natural Language Processing (NLP), for example, audio tagging, opinion mining, as well as dependency parsing.

Keywords:
Word Prediction, Naïve Bayes, LSI, Named Entity Recognition (NER), N-Gram, Sequence Prediction, Neural Network, L2AWE (Learning To Adapt with Word Embeddings), LSTM, RNN, Augmentative and Alternative Communication. sentiment similarity weight (SSW), XLM, XLMR, statistical language model, RoBERTa, Stupid Backoff Algorithm

Introduction:
Language models (LMs) are used in a variety application involving natural language processing (NLP), such as language modeling, machine translation, and speech recognition [1]. Many sequence learning tasks, such as machine translation, language modeling, and question answering, use recurrent neural networks (RNNs), such as long short-term memory networks (LSTMs), as a key building component [2]. LMs provide good control over a wide range of languages. The majority of these models learn language by maximizing each word's in-context probability in their training corpus, which is commonly done with a self-supervised aim [6]. LM assigns the probability distribution of the text based on available text information and is a fundamental task in natural language processing (NLP) [2].

A common method is to utilize a network to encode the context into a fixed-size vector that is then assigned a category probability distribution over the next token [3]. Models that develop astonishing powers of psychometric prediction and language in general have been created using simple corpus probability matching [7]. Unsupervised distribution estimate given a series of instances (x1, x2,..., xn) each made of variable length sequences of symbols is how language modeling is commonly described (s1, s2, ..., sn) [7]. Because language has a natural sequential ordering, it is common to factorize joint probabilities over symbols as the product of conditional probabilities. In theory, language modeling learns tasks without requiring explicit supervision of which symbols should be expected as outputs [7]. Language modelling is substantially aided by a thorough understanding of sentences. In comparison to previous architectures, with significantly greater accuracy, neural networks can be trained to predict words from their context. Both recurrent neural networks and non-recurrent attention-based models have been demonstrated to do this [10]. Proposals have been made for testing the models on subsets of the test corpus, where effective word prediction is based on a right interpretation of the sentence structure, in order to gain a better understanding of their successes and failures [10]. Recurrent neural networks (RNNs) are a common framework for modeling sequential data and are utilized in a variety of NLP tasks, including language modeling, question answering, and machine translation. RNNs obtain the conditional probability from each hidden state in language modelling [3]. The sentences, on the other hand, are modeled sequentially using RNNs, which do not effectively utilize structure information. Learning the latent hierarchical structures in RNNs can lead to better language models [3]. Researchers employed RNN to capture sequential information based on the sequence properties of texts. To address RNN's deficiency, LSTM was used [4]. Scientists have employed a flexible framework to increase LM performance [11]. They focused on two forms of advanced NLMs: LSTM and Transformer-based networks, both of which have lately attracted a lot of interest in the NLP field. In their system, there are three stages: (a) a context representation learning stage for encoding the variable-size context into a dense representation, (b) a sense-labeling stage for inferring a probable sense based on the learned representation, and (c) a multi-sense LM (MSLM) learning stage for learning multi-sense representations with the inferred senses [11].

The goal of Named Entity Recognition (NER) is to recognize named entities in a text and classify them into pre-defined domain entity types as people, organizations, and places [5]. Most existing NER systems use generic entity type classification schemas; nonetheless, even for human specialists, comparing and integrating (more or less) diverse entity types across different NER systems is a difficult task [5]. The L2AWE (Learning To Adapt with Word Embeddings) supervised approach for adapting a NER system trained on a source classification schema to a target classification schema has been developed by researchers [5]. One of the most significant Information Extraction (IE) operations is discovering entity mentions, which are text fragments that imply real-word things, from unstructured text and classifying them into entity categories according to a predefined classification scheme. The problem was chosen to extract the hidden treasure of knowledge that we receive through textual data generated by online interaction. To use such useful insights for decision-making, the unstructured data must be processed using NLP techniques to extract actionable insights in a machine-readable format. Extracting crucial information from user-generated content in the form of entity mentions, events, and relations is critical for knowledge discovery from natural language text [5]. The hypothesis that embedding representations of named entities can boost the semantic significance of the feature space used to perform source-to-target domain adaptation is confirmed. The problem was chosen to extract the hidden treasure of knowledge that we receive through textual data generated by online interaction. To use such useful insights for decision-making, the unstructured data must be processed using NLP techniques to extract actionable insights in a machine-readable format [5].

Living in the age of social media, text-based communication is increasingly important. Messages are used to communicate in every part of life. It is critical to effectively frame messages so that the message's meaning is conveyed to the point. After the correct sentences have been outlined, words from these sentences can be chosen to suggest emojis [8]. At this point, emoticons play an important role. The meaning of the message can be easily conveyed with the use of accurate emoticons. Emojis are an essential aspect of communication. During a conversation, it is utilized to express feelings. It can be quite handy to create a system that can propose emoticons based on the text provided. It may be used to quickly and effectively express emotions [8]. It can be used to anticipate the emotion in a sentence and emojis can be predicted accordingly while dealing with the semantics of the phrase. Typing each and every word to form a phrase is also a time-consuming activity that can be made considerably easier with the help of word prediction algorithms [8]. As a result, integrating two models, namely, word prediction and emoji recommendation, will enhance textual communication [8]. By looking at the statistics of connection circle, a preliminary survey of emojis reveals that a small number of people prefer to use emojis [9]. Two surveys were conducted to find out how people use and perceive emoticons. Users may choose emoticons based on their tastes and qualities, and they may have diverse meanings of the same emoticon at times [9]. Study's sample size is modest [8], more people are needed to confirm the findings. Building a system to use supplied predictions to complete a sentence or partial words, then utilizing the created sentence to recommend emoticons to make the sentence more appealing. When predicting the finish of a sentence, many aspects must be considered. Because user's input may be incorrect, the most comparable terms to the incorrect input should be anticipated, and the entire text can be made more appealing using an emoticon recommendation model [8].

Architecture:

We came across several architectures while reading research papers. The framework was divided into three stages [11]. During Stage 1, a single-sense LM learns context representation vectors (hcontext (ws)) from word tokens. Stage 2 involves splitting the acquired context vectors of each multi-sense word in a training corpus into numerous new word-sense pairs using an unsupervised clustering algorithm [11]. Finally, from word-and-sense tokens, a multi-sense LM learns context representation vectors, wm, in Stage 3 [11]. Datasets were taken where n gram model and cosine similarity were used for emoticon projection [8]. The structure of a one-layer RSD-LSTM is illustrated [3]. Between the inputs, convolution is used [3]. The links between states and gates are represented by hard arrows, while the connections between distinct timesteps are represented by dash arrows [3]. Location, Timestamp, Emotion, and Professional Relationship are the four contextual variables are mentioned [12]. The system's five key processes are data collecting, text preprocessing, word embedding, model training, and deployment [13]. Text data is sent into the algorithm, and the intended output is a few possible following syllables. For vector representation of knowledge, the preprocessed text is input into the word embedding. For training, the vectorized text sequences are supplied into the LSTM model [13]. With all of the ways we've tried, we've come up with our own simple architecture.

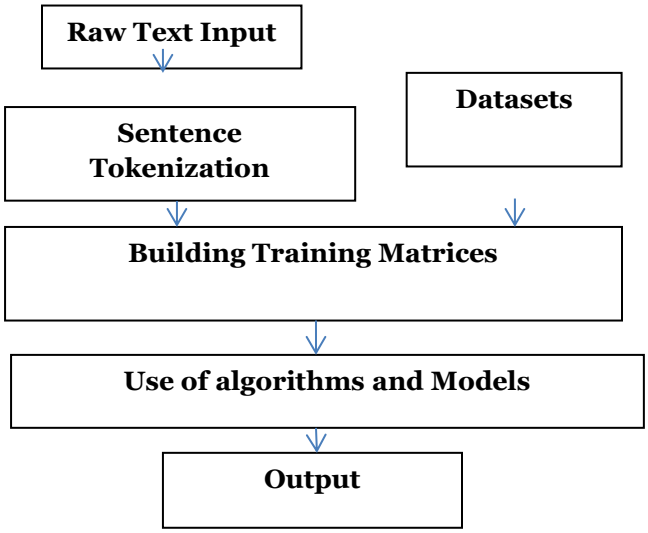


Figure: Architecture for next word prediction

In our model, we have created two sections. First, if user is giving the input then sentence tokenization is needed after which building the matrices. The technique of breaking text into individual sentences is known as sentence tokenization. Second, if we are directly working on the dataset then we can directly process in building the matrices. Then after different kinds of modules and algorithms can be used to train the model. Finally, required output is shown i.e. predicting the next word.

The most common parameters for evaluating word prediction models

- Accuracy:

Our goal with the precision metric is to determine how near a measured value is to a known value. As a result, it's most commonly employed in classification tasks where the output variable is categorical or discrete.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{False Positives} + \text{True Negatives} + \text{False Negatives}}$$

- F1 Score:

Precision and recall are two measurements that are mutually exclusive and have an inverse connection. We'd utilize the F1 score to integrate precision and recall into a single statistic if we're both interested.

$$\text{F1 SCORE} = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

- Precision:

Precision would be used in circumstances when the precision of the model's predictions is important. The precision metric would inform us how many labels are truly designated as positive, matching to positive cases indicated by the classifier.

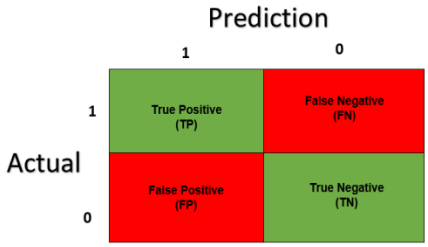
$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives + False Positives}}$$

- Recall:

The model's recall is a measure of how well it remembers the positive class (that is, the number of positive labels that the model identified as positive).

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}}$$

A confusion matrix is created by combining the four characteristics mentioned above. The confusion matrix is a valuable tool for machine learning classification problems, with the main purpose of visualizing a machine learning model's performance.



Popular Datasets used for word prediction

- Project Gutenberg is a massive collection of unabridged books available in a variety of languages.
- The Current-Day American English Standard Corpus at Brown University. This collection includes a substantial number of English phrases.
- Google's 1 Billion Word Corpus

Our Contribution:

On reading through articles, journals we were astonished to find that fair amount algorithms can be used for the word prediction. Prediction is basically finding what is coming next. Hence, predicting words all in all is not a cup of tea. Especially, languages like Urdu, Arabian, Kurdish, Chinese are extremely tough to predict than that of English. There are grammatical ambiguities in all languages. Unsupervised phrase segmentation experiments throughout languages have time and again proven that English is less difficult to phase than different languages. [65]. In English, for example, the word "you" can refer to a single individual (second person singular) or a group of people (second person plural). The word "you" might be misleading when pronounced without a modifier (such as "you all"). Another example is ancient Hebrew, where verbs are either complete or incomplete. This can lead to uncertainty, particularly when translating into a language with past-present-future tenses or a distinct verb tense system. Ancient languages, especially those that are no longer spoken, have a higher percentage of terms whose meaning is unknown, making things more unclear. The Hebrew word "sheol," for example, is never fully defined

in biblical usage, hence its meaning is ambiguous. Similarly, the meaning of the Hebrew term "selah," which appears in the Psalms, remains uncertain. We can compare these terms to comparable words in other Levantine languages to try to figure out what they mean, but it's still a guess. The more ancient the language, such as Sumerian, the more ambiguous the words, leading in difficulty on predicting the word. In the realm of natural language processing (NLP), word prediction is one of the most fundamental tasks for language models, which employ a language model to determine the joint distribution of natural language word sequences. The possibility of the characters or words that follow is regularly predicted using a few characters or phrases. As a result of the development of profound learning innovation, language models based on neural organisations have been highly embraced in the field of NLP..

- **Process and difficulties:**

Auto-completion and sequence prediction can be a core component in query-related systems or recommendation-based interfaces to integrate the system with human engagement. It betters the user interaction by guessing the word(s) that the user wants to type. There are numerous techniques to improving the solution. Previous research in this area has provided insight, however the prediction-based method for Bangla does not meet that standard. For Bangla word prediction, some studies recommend using N-grams with deleted interpolations and a back-off model. Several implications of the proposed framework can be on any autocompletion-based feature in Bangla, keyboards, chatbots, recommender systems, Q/A based systems, etc. hybrid approach used, particularly the sequential model, is primarily evaluated for sequence prediction. Developed model completed training with a maximum accuracy of 84 percent. The first and second datasets were both used in this study. Though process interpolation or backoff might have improved the rate, we hadn't looked at it during the training phase. We keep it on our to-do list as a major project. One is known-sequence input or user input that the model has already seen during the training phase. Another example is the case of out-of-vocabulary (OOV). Scientists attempted to present their performance of the model as well as its efficiency. For one input sequence, the sequential model predicts some sequences. These sequences are entered into the n-gram section of the model to sort out the best sequence among them based on the best grammatical pattern, such as the correct use of a specific word or the sequence's most often used word. We switched to the RNN (Recurrent Neural Network) model [19], focusing on the sequential model, which is well-known for machine translation, sequence generation, and other applications. When it comes to constraints, it hasn't been a perfect circumstance for us throughout our trip. To improve the accuracy of the result, we had to try a few different approaches.

Language modelling is one of the most frequent approaches to problems with natural language processing. Neural language models became popular as a result of the transfer learning approach (using a model trained for one task to another task). However, research into Turkish language models utilising neural language models has yet to reach acceptable levels. To compare statistical and neural language models, the prediction of a word that was closed randomly in a sentence was utilised [64]. It was revealed that neural language models had a better success rate [64]. The efficacy of neural language models and statistical language models was evaluated in order to predict the missing word in a Turkish phrase. Markov-based approaches were employed as statistical language models. We used both pre-trained models and models that we trained using our own datasets as neural language models [64].

Everyone stays linked on the go, usually through chat and text messaging, thanks to digital communication, which has become a part of everyday life. Many anticipated and recommended terms are supplied by the keyboard to aid users in typing while chatting and texting. These keyboards' technologies do not take into account the users' context or the mutual situation of both users in a chat. The mutualization of the contexts of two users engaged in a discussion is referred to as mutual context.

Mutual context is described as the mutualization of the contexts of two users participating in a discussion. Context is defined as the physical and psychological state of a user, and mutual context is defined as the mutualization of the contexts of two users involved in a conversation. For users' mutual contexts [55] are used when predicting words, the accuracy of correctly guessing the next word in a conversation improves. Incorporating the mutual context of participants in a normal chat using a simulation tool yielded positive results. Using C++ and the standard template library, a rudimentary simulation tool based on perspectives (collecting a user's contextual values) was built for generating results, with the crucial fact being in the word selection section. [56] There are a lot of good word suggestion and word prediction techniques out there, but nearly none of them are immediately usable in conversation. Experts utilised a three-stage probability [57] tree to predict the following word. So, initially, the researchers accepted the four contextual data (Location, Timestamp, Emotion, and Professional Relationship) as direct user inputs, and then the users began typing. After the contextual values were collected and enumerated in the back-end, the context was generated. The context value is then used to locate or construct a table, which is subsequently used to store new words or recall previously stored ones in order to predict conversation terms. The words available are picked based on their likelihood ranking. After two preceding words, the projected word is the one with the highest possibility. Researchers gave this simulation tool to a set of participants, along with a questionnaire to fill out in order to review the experience while using the software..

In digitalized conversation emoticons are also important. It's used to express emotions throughout a conversation. Creating a framework that can suggest emojis based on the content provided might be really useful. It can be used to predict the feeling in the statement and emoticons can be predicted based on it while controlling the semantics of the text. Composing every single word to complete a phrase is a time-consuming task; but, with the help of word forecast models, this task can be greatly simplified. Sentence outlining may be completed with ease using an exact word prediction model. After the appropriate sentences have been outlined, words from these sentences can be chosen to suggest emojis. The framework will make use of the bigram trigram model to anticipate words and assist in sentence formation, which will also be added to the emoji idea model. Emojis are used in a variety of ways by different people. The manner of utilisation by diverse people can be predicted with the help of bunch division. Emoji predictions are made using recurrent neural networks. The principal layer, the hidden layer, and the last layer would all be part of this methodology's repeated neuronal organisation [22]. At the point when the client utilizes any emoji, its arrangement would be recorded and in this way one of the neurons in the main layer would hold the worth of that emoji. In light of the occasions the emoji is utilized in that specific arrangement, the weightage of that neuron would increment. Numerous such neurons would be holding various qualities or various groupings in which emojis were utilized. To anticipate words in a succession, a bigram and a trigram module is created in python. The bigram module registers the probability of a word after a given sequence. This is cultivated by saving every one of the expected words in the corpus, inside a variable (in python). The count of this bigram is a key value pair in a hashmap. By dividing the value(count) by the total number of times the given word appears in the corpus, the likelihood can be calculated. In addition, the trigram module is used as a hashmap, with prospective words appearing before a grouping of words (two words) with their own check. Hashmaps can be used to perform faster lookups. In real-world typing, clients frequently make mistakes, necessitating the use of a skilled typing aid. This is accomplished by employing the Minimum Edit Distance concept, which aims to set expectations for what the client must input. This is accomplished by the use of dynamic programming [24], which determines the smallest amount of expansion, deduction, and replacement operations required to transform a single word into something extremely similar to another.

With the quick and persistent advancement of the Internet and the approach of web-based media, the measure of unstructured text-based information created by the social cooperation's among individuals has turned into an immense secret fortune of information. So, to take advantage of such significant experiences for dynamic purposes, text-based information should be handled to remove noteworthy bits of knowledge in a machine meaningful structure by utilizing Natural Language Processing (NLP) procedures. L2AWE (Learning To Adapt with Word Embeddings) is a guided approach that aims to modify a NER framework based on a source order pattern to a specific goal one [34]. In particular, findings agree with the idea that incorporating depictions of named substances might affect the semantic significance of the element space utilised to play out the change from a source to an objective region. The goal of Named Entity Recognition (NER) is to recognise named components in a text and categorise them into pre-defined space element kinds such as persons, associations, and regions. Although the majority of present NER frameworks use nonexclusive substance type characterization compositions, the examination and coordination of (pretty much) distinct element kinds amid multiple NER frameworks is a puzzling problem for human experts [35].

The Kurdish language belongs to the Indo-Iranian language family. Kurdish is the language spoken by Kurdish people. The suggested application is designed for Kurdish speakers (of the Sorani and Kurmanji dialects). When a user types a word into this system, the user is given a list of five words to choose from. Based on the preceding written word or words, the suggested method will recommend five words. These ideas are based on a Kurdish text corpus and employ the N-gram model. The following are the contributions of this work: It creates the first Kurdish corpus text, which contains over 500,000 words in both languages. It overcomes the difficulty of reading Kurdish letters in RStudio, which is a problem because the programme doesn't support them ((.(ه، ۆ، ێ، ڵ، ژ، پ Kurdish is a Kurdish dialect. The Kurdish language has its own letters and characters, much like any other language. Indefinite nouns can have one of the following endings: a vowel for singular or plural, or a consonant for plural. When the stem verb is preceded by "da" in the past progressive, the clitic is appended to "da," as seen in the following example with the verb. Unigrams, bigrams, trigrams, four-grams, and five-grams have been created in the proposed system for Kurdish Sorani and Kurmanji. The word prediction algorithm for the Kurdish language was built using a big corpus. [61] To forecast the likely terms, the Stupid BackOff algorithm (SBO) is used. Because of the vast corpus or dataset and the fact that SBO is great for word prediction, the algorithm performs better. Because SBO estimates the score rather than the probability and does not need to standardise the score or probability, it works better with a big dataset, such as the one used for the Kurdish language. SBO uses a backoff score that may be written as Lambda and equal instead of discounting since it uses proper frequencies instead of discounting. For the Kurdish languages of Sorani and Kurmanji, a word(s) recommendation system has been developed. Because this is the first time prediction models for the Kurdish language have been developed for both the Sorani and Kurmanji dialects, it was more difficult to generate this corpus and these grammes than it would be for the English language. More than 500,000 words in Kurdish Sorani and Kurmanji have been added to a new Kurdish text corpus [62].

In Dzongkha, a word can be made up of a single syllable or many syllables. A single syllable and a multiple syllabic word need six and twenty-two keystrokes, respectively. The majority of the syllables and words take multiple keystrokes. The goal of this research was to create a syllables prediction system that would cut down on keystrokes and typing time. Elsevier B.V. produced and hosted the video on behalf of King Saud University. Seven keystrokes are required to type a single syllable word, which consists of four letters (ཀ, ད, ཨ, ད) and a vowel (ོ). The syllable prediction method cuts down on typing time while also encouraging people to write. The goal of this research is to offer a method for predicting

syllables in Dzongkha using n-grams, word embedding, and Long Short-Term Memory (LSTM). Data capture, text preprocessing, word embedding, model training, and deployment are the five primary stages of the system. The system's input is text data, and the expected output is a few likely predicted syllables. The preprocessed text is fed into the word embedding for vector representation of knowledge. The vectorized text sequences are fed into the LSTM model for training. To improve accuracy, the word embedding and LSTM hyperparameters are fine-tuned. The model is trained until it achieves the desired accuracy. The model is saved for deployment once the expected accuracy has been achieved. The Django web framework, along with Bootstrap and JavaScript, is used to deploy the model.

Urdu is an Indo-Aryan language spoken by around 163 million people globally; nevertheless, despite its extensive use, due to a paucity of corpora and datasets for performing several computational tasks, the language has remained under-resourced [31]. Word embedding is one answer to this problem. Embeddings are the vector representations of words. It's used to translate words from a text into lower-dimensional representations, as well as to understand some key properties about the text data. The task of creating a distributed representation of words in vector space is known as word embedding generation. To execute natural language processing tasks, word embeddings capture both syntactic and semantic information. Word embeddings have gotten a lot of interest in recent years because of their intrinsic usefulness in Natural Language Processing, such as audio tagging, sentiment analysis, and dependency parsing. Despite recent advances in word embedding mappings, there has been very little research dedicated to Urdu word embeddings, where words with similar meanings align to comparable vector spaces and are separated by a large distance. The Word2vec model was utilized to generate Urdu word embeddings in this research report. It's made with the Word2vec model's Skip Gram [32]. This approach generates vectors by examining the context of words in sentences. On vector space, words with same context appear near together, and words with dissimilar context appear far away. Unsupervised neural networks are used in this Skip Gram model. On the collection of data, this model does both syntactic and semantic analysis.

Natural languages have a hierarchical structure that is determined by complicated syntactic trees that show the internal syntactic structure of a sentence: a sentence is made up of several phrases, and different words make up phrases. The link between the parts of a sentence in natural language is generally tree-like: words, phrases, and clauses construct a sentence hierarchically, and syntactic structure is created by the dependency between different parts. Understanding natural languages requires such a complicated tree-like structure. Recurrent neural networks (RNNs) [49], on the other hand, model languages sequentially and fail to encode a hierarchical syntactic relationship adequately, resulting in underperformance on comprehension-based tasks. In this paper, researchers developed a new neural language model termed relative syntactic distance LSTM (RSD-LSTM) to dynamically represent syntactic structural dependencies [50]. The relative syntactic distance between sentences is computed using a convolutional neural network to represent the degree of dependency between words, and the relative syntactic distance is used to modify the LSTM gating mechanism. There was an introduction to direct link between concealed states to combine high- and low-level syntactic characteristics. WikiText-2 (WT2), Penn Treebank (PTB), the PTB dataset, which contains about 1 million tokens, has been pre-processed to remove capital letters, numerals, and punctuation. The wikitext-2 dataset is roughly twice as large as the PTB dataset and is derived from a Wikipedia article. Baseline model was the AWD-LSTM, which produces state-of-the-art results on the PTB and WT2 datasets [51]. The recurrent neural network (RNN) can recognize the grouping association between words in a sentence and apply relevant semantic data to the current circumstance, improving the sentence's semantic significance. The convolution part of CNN is presented into the language model of succession expectation task to fully mine the nearby element data between word groupings and work on the model's prediction. It used convolution of varied window

sizes to look at the convolution activity of contiguous word successions, improving the impact of neighboring word groupings on the expectation targe. The MGU unit uses residual connection to address the problems of vanishing gradient [28] and network degradation caused by a rise in the number of organization layers, allowing the model to do more in-depth training when the network depth was quite deep. The MCNN-ReMGU model can successfully acquire global and neighborhood highlight data between word arrangements, as well as uncover the word grouping relationship in sentences, hence enhancing the model's word expectation capacity. The findings of this paper's analysis can be used to provide specialized assistance for automated writing, text content information, and other projects. This research introduces the MCNN-ReMGU model for natural language word prediction, which is based on multi-window convolution and residual connected MGU network combined with information regularisation innovation. The effectiveness of multi-window convolution and the residual connected MGU network [29] in removing high-dimensional elements between locally nearby words and feature information between word sequences is proven using the PTB dataset [30]. Simultaneously, experiments reveal that the surviving association with the MGU network not only solves the vanishing gradient problem and network degradation, but also fully learns the long dependance relationship between word successions. Overall, the test results suggest that the proposed MCNN-ReMGU outperforms standard approaches in the exhibition of the word expectation task. He also offered a strategy that allows state-of-the-art techniques to compete.

In a range of natural language processing (NLP) applications, such as language modeling, machine translation, and speech recognition, LMs determine the likelihood of word sequences and are designed to offer high-probability sequences that are both semantically and synthetically meaningful. [52] To achieve strong performance, LMs must accurately record the relationships between words and phrases in word sequences. A context representation stage encodes the variable-size context, a sense-labelling stage uses unsupervised clustering to infer a plausible sense for a word in each context, and a multi-sense LM (MSLM) learning stage learns the multi-sense representations. [53] For analyzing MSLMs with varied vocabulary sizes, experts developed a new metric called unigram-normalized perplexity (PPLu), which is also known as the negated mutual information between a word and its context information. On the shift in vocabulary size, there is also a theoretical PPLu verification. Researchers compared their findings to previously published results using the PTB and Text8 datasets. PPL values were calculated using both PPL and unigram probability, whereas PPLu values were produced using both PPL and unigram probabilities. LSTM LMs with word embedding dimensions of 200 and 650 were used to achieve the findings. Although some published results, such as Sum– Product Net, do not employ LSTM, they perform significantly worse than LSTM. As a result, they chose the LSTM-based LMs as their single sense LM. In terms of the indicated PPLu, multi-sense LMs with nine senses outperformed single-sense LMs.

Cloze Distillation shows how altering training incentives away from corpus probability and toward psycholinguistic errand-based modeling can result in better cognitive and linguistic models [40]. In any case, there are a few potential clarifications for the impact Cloze Distillation has on language model performance, given that a few of our models anticipate perusing times beyond the cloze information acquired in Provo. The author has found a close relationship between perplexity and syntactic generalization, which adds to a growing body of evidence and suggest that while optimizing a corpus probabilities can create somewhat psycho-linguistically enabled language models There may be dissociation between human expectation and corpus probability One is that, although being under-sampled, the Cloze errand generates data that is a more reliable representation of the assumptions generated from human reading and hence ready to direct the models toward a more human-like arrangement of assumptions [41]. In the Provo corpus data [42], which was collected from 84 native speakers of American English, cloze probabilities are highly predictive of reading times. This study adds

to earlier research that suggests that the length of time it takes to read a word is related to its predictability. For psycholinguists interested in treating LMs seriously as candidate models of human language preparation and for regular language handling analyzers interested in figuring out and expressing bits of knowledge from human sentence preparation, Cloze Distillation, for example, provides a path forward.

SCDE is a sentence cloze dataset developed by humans and derived from Chinese public school English exams [43]. The job needs a model to use distractors created by English teachers to fill in many holes in a passage from a shared candidate pool. They've released their own dataset, SDCE, which is a human-created phrase cloze dataset gathered from school examination contexts with cloze gaps that must be filled with a produced word or by selecting one from a list of options. PMI simply indicates how frequently a word pair is to appear in consecutive sentences. It ignores internal sentence structures and the relative placement of words within their individual sentences. Fine-tuning BERT yields the best results among other models; however, it still falls short of human performance [44]. Only one-third of the blanks could be solved using unsupervised models. Surprisingly, the unsupervised models outperform PMI2 and COHERE [45]. To distinguish between the ground truth response for a single blank and another choice that is a ground truth answer for a neighboring blank, COHERE relies solely on syntactic regularities. SCDE may, among other things, encourage the development of more advanced natural language comprehension models.

The neural network is trained to guess the term, but only after it has gained a thorough knowledge of the context. The verb must agree in number with the subject (singular or plural). In order to do so, LM must first comprehend the sentence's context. For word prediction, CLAMS data has been introduced [46]. Lstms are statistical models that predict the likelihood of syllables, to put it another way, the likelihood of the next word in a sentence given the preceding ones. His study on BERT and mBERT [47] suggests that the latter shows signs of learning in several languages, that it learns these types of rules faster in some languages than others, and that its sensitivity to syntax is weaker than monolingual BERT [48]. While more languages must be investigated to determine whether or if this concept is resilient, it is compatible with.

In mainstream spoken language processing systems, automatic speech recognition (ASR) and a specialized natural language module are frequently integrated in a cascade [25]. The usual language preparing module in such a framework may be a text recovery module, an inquiry noting module, a machine interpretation module, and so on. The purpose of this paper is to translate speech into text (ST). The ASR acknowledgment results influence the appearance of any of these regular language handling modules. As a result of incorrect ASR acknowledgment findings, downstream natural language preparation modules make incorrect assumptions, resulting in debased execution. A low-WER ASR module is necessary to prevent such cascaded error propagation. Another option is to avoid error propagation by using a direct end-to-end model or ensuring that enough data is available to train end-to-end models. End-to-end models are easier to build, have less inertia, and, in some situations, provide better execution than cascaded systems. To achieve unsurpassed ST execution, one must either avoid error propagation in cascaded systems or ensure that enough data is available to train end-to-end models [26]. Consistent word depictions, also known as word embeddings, are used in both scenarios. During training, speech models convert speech signals to matched text and understand the relationships between speech and text. Speech models are created using the data and relationships communicated by word embeddings. It allows speech models to refer to additional semantics in the output text, allowing them to capture precise meaning. The sequence-to-sequence ASR models are taken into consideration, with an autoregressive decoder being used to anticipate the record relating to the information discourse. The

decoder is given the task of predicting word implanting in this case. Due to their various constraints and creating discourse record sets, end-to-end ASR frameworks accomplish execution that is almost equivalent to (or startlingly better than) regular ASR. The objective language decoder decodes using the secret states provided by the discourse encoder and the source language decoder. For ASR and ST [27], this research recommends using pre-trained word embeddings, which requires less effort and does not require enlarging model boundaries. Word embedding regularisation in ASR permits idle decoder provisions to be strongly correlated with the appropriate objective word implanting, which may subsequently be utilised to improve ASR yield with entangled decoding. It helps the cascaded device of translation. Word embeddings are used as a transitory in executing many operations in the end-to-end architecture, and it helps with interpretation quality. Furthermore, with the provided methodologies, a pre-trained strategy is found to bring additional enhancements. The results suggest that learning acoustic-to-semantic transitions is possible.

- **Model performances:**

For the Bangla language, a hybrid framework has been proposed. The proposed approach includes autocompletion, which uses the trie data structure to complete a word from a user-provided prefix. A hybrid implementation of the grouping to-arrangement LSTM model and N-gram yields [20] a superior sequence prediction model. This work's evaluation shows how their model beats several n gram-based Bangla prediction methods [21]. It was that dataset A has better accuracy and lesser failure rate than the dataset B. Auto-completion and sequence prediction is catchy features for any type of recommendation system. Sequence prediction can be a necessary issue for writing assistance systems. We believe this work will help these systems and encourage future researches in a very impactful way.

For Kurdish language framework uses the N-gram language model, which is reasonable for enormous datasets and valuable for computing frequencies of words rather than complex probabilities. 17.4% precision, 46 ms reaction time score when a forecast isn't found, for example, from the quad slam to the trigram. The Kurdish language is not at all like the English language and a few issues were confronted when fabricating the framework, the N-gram language model [63] has an exactness of 96.3%. A Kurdish text corpus with in excess of 17 million words exists, there were not satisfactory assets to utilize it in the proposed framework; an extremely incredible PC is expected to process a text corpus of that size.

The single-layer Long ShortTerm Memory with 128 memory cells acquired the best preparing precision of 78.33%. In the word forecast from the given arrangement of current expressions of the Assamese language was proposed utilizing LSTM. They have accomplished an exactness of 88.20% and 72.10% for message and phonetically transcripted the Assamese language [58] separately. It has been seen that one-layer LSTM with 128 cells gave the best precision of 78.33% and the least deficiency of 0.7110. The preparation exactness acquired was 78.33% with a mistake pace of 0.7110. The preparation exactness of the model was 78.33% with the most reduced mistake pace of 0.7110

A sample of ~675,000 words were taken from the test dataset. The Bigram model accurately predicted the next word 54% of the times and the trigram model predicted the next word 59% of the times. The proposed system has the capacity to autocomplete and forecast the next word based on past words, increasing typing speed and efficiency while also making the system more intelligent [23]. The technology will boost the number of emoticons used in conversations. The manual effort of selecting emojis is made easier by a prediction engine.

For statistical language models, a Markov-based procedure was utilized. Freely accessible pre-prepared models and models made without any preparation were utilized to make neural language models. As per the discoveries, neural language models beat measurable language models. These discoveries are in accordance with those of different examinations on the English language [64]. The achievement of neural models over measurable models is because of the way that for factual models to anticipate a word in the test sentence, the word to be assessed should show up in the preparation set with the N word around it [64]. This is far-fetched on account of profoundly unique successions like language. By encoding words/arrangements in a consistent space, neural language models give answers for this test [64].

A three-layer RSD-LSTM model is utilized in the model. The outcomes show that RSD-LSTM further develops perplexity by 1.82 and 2.03 when contrasted with current top techniques on the Penn Treebank and WikiText-2 datasets, individually. Scientists have plainly fostered the preferred model over every one of the past ones. On all datasets, the model beats gauge machine interpretation models just as cutting-edge language models.[3] The findings suggest that the model can be used to derive the hierarchical structure of real languages. Furthermore, the dataset visualization shows how effectively the proposed relative syntactic distance works for interpreting texts. The Text8 dataset is used to pre-train all models. \ The Matthews connection coefficients and the arrived at the midpoint of upsides of the relationship coefficients are accounted for CoLA and STS-B, individually. For any remaining undertakings, the precision of each assignment is accounted for, just as the normal worth [3].

The inherent ambiguity in Japanese simply indicates that a given sentence can be segmented in various ways. The fact that these repeated segmentations result in systematic segmentation errors yet to be established. To do this, we offer a supervised segmentation algorithm that enumerates all potential utterance segmentations using the gold lexicon and selects the segmentation with the highest probability [65]. This approach produces a segmentation F-score of 0.99 for English and 0.95 for Japanese in CDS data. In ADS, English has an F-score of 0.96 and Japanese has an F-score of 0.93. These findings demonstrate that lexical information combined with word frequency removes nearly all segmentation errors in English, particularly for CDS [65]. Even while the scores in Japanese remain astonishingly high, the lexicon alone is insufficient to eradicate all errors. To put it another way, English is still easier to segment than Japanese, even with a gold lexicon [65].

The small size multi-sense LM, specifically, beats both enormous size single-sense LMs.[54] Furthermore, utilizing a similar organization plan, the upgrades from single-sense to multi-sense LMs are huge, with a 31.2 percent decrease from 0.1527 to 0.1051 and a 24.0 percent decrease from 0.1336 to 0.1015 for the 200 and 650 implanting aspects, separately. Created model for Urdu language will help with making a thick word vector portrayal of Urdu words that can be utilized to prepare word vectors. The outcomes showed that the proposed technique can be used to further develop existing word installing strategies [33]. We can use these embeddings to estimate feeling extremity later on. Feeling extremity will help opinion examination on multilingual datasets. The assessment results show that the recommended traffic-related record arrangement model using the SSW has a sensible f-proportion of 0.907. It is possible to extricate profoundly applicable traffic-related reports from unstructured information in an assortment of settings utilizing the recommended model, order the recovered records, identify creating risks, and advise clients regarding potential traffic chances utilizing the proposed model [39]. Proposed MCNN-ReMGU gives the least (perplexity) PPL esteem contrasted with the models having similar number of stowed away units and a similar organization profundity. Additionally, for a similar secret unit condition, the PPL worth of the model proposed in this paper diminishes as the quantity of organization layers increments.

Collation of base papers

paper	Datasets	**Algorithms**	Scope	Application
[3]	Penn treebank, WikiText- 2	Rnn, Lstm	Relative syntactic distance	Machine Translation
[11]	Semcor, PTB,Text8	CBoW,LSTM, BiLSTM, XLNet	Not mentioned	NLP
[12]	Wordnet	Three stage Probability Tree	Mutual context	Real time such as android keyboard
[5]	#Microposts201, #Microposts2016 and W-NUT17	MMCM, TUCM, FMCR	NER	NLP
[19]	Sentiments Dataset	Cosine Similarity, GloVe, FastText	Text mining, Risk Detection	Sentiment Similarity Weight (SSW)
[20]	SST, Roman-Urdu Dataset	LSTM, Bi-LSTM, Tree-LSTM, Skip-gram Algorithm	Sentiment Analysis	NLP
[8]	emoji depiction and their individual unicodes,	GloVe vector. cosine similarity	Not mentioned	chat applications, web applications
[27]	LibriSpeech, Augmented LibriSpeech,Fisher Spanish Corpus,NMT	Langtgt, Lang src, Multitask learning, pre trained word embeddings, cosine distance similarity	Improve understanding in order to perform more activities using spoken language processing.	enhance ASR output with fused decoding
[28]	PPL, PTB, WT2	CNN, MGU, MCNN, ReMGU	remove provisions between word successions	computer vision applications
[6]	Provo dataset	5 gram LSTMGPT 2 Transformer XL	Improve reading time prediction with Cloze Distillation	accurately infer next-word
[66]	DailyMail	PMI	future models to improve to match human performance.	INFST feeds two encoders with a blank and a candidate.
[67]	CLAMS dataset	TreeTagger5	alterations in the architecture of neuronal LMs (eg. better	multilingual LM

			handling of morphology)	

The syntactic links between consecutive pairs of words in a phrase are represented by a syntactic distance model [3]. When the syntactic distance between two words is considerable, it suggests the two words might be in two separate sentences. They are, nevertheless, inside the same sentence. The height of the rock bottom common ancestor between two adjacent words is an easy description of syntactic distance. Furthermore, syntactic distance does not accurately reflect the degree of reliance between two syntactic routes [3]. To avoid the ambiguity of word embeddings, researchers have divided the meanings of multi-sense words based on their surrounding context. Many previous efforts have deliberately integrated a word with multiple meanings into various independent vectors for distributed representation models such as continuous bag-of-words (CBoW) and skip-gram (SG) models. Using neural LMs, some models created explicit sense vectors. These models, however, have a number of flaws [11]. While the multi-sense embedding's performance enhancement was restricted to prominent for simpler LMs, not so for complicated LMs like LSTM [11]. Emoticon is also important for the users in chatting. While predicting word for sentence completion, a lot of parameters have to be considered. Given input by the user might not be right, most similar words to the wrong information should be predicted and to make the text really engaging with the assistance of emoji suggestion model this can be accomplished [8]. For solution to this problem, developed system is divided into two parts, the first of which predicts words and the second of which predicts emoticons. Brown corpus is used in the word prediction model which includes information from a number of sources, including news stories, novels, and other literary works. Through bigram and trigram implementation, the goal of word prediction has been met. This information is transmitted into the second system which is an emoticon proposal that makes use of the 'GloVe Vector'. For the Urdu language, Word2vec outperforms for Long Short Term Memory (LSTM), Bi-directional Long Short Term Memory (Bi-LSTM), and Tree-Long Short Term Memory on the Urdu dataset (Tree-LSTM).

paper	**Supervised or Unsupervised**	**Similarity**	**Metrics**
[3]	Supervised	RSD-LSTM	BLEU
[11]	Both	Group depending	PPlu
[12]	Not mentioned	Not mentioned	Absent
[5]	Supervised	Baseline-Deterministic (BL-D),Baseline-Probabilistic (BL-P1),Conditional Random Fields (CRF)	String Similarity
[19]	Supervised	Manhattan-distance-based similarity measure	Absent
[20]	Unsupervised	GloVe, FastText	STD
[8]	unsupervised	CD	Absent

[27]	Both	Recognition Error Rate with Skip-grams, Comparison with Semi-Supervised Methods	Translation metrics
[28]	Not mentioned	RNN, LSTM, GRU	Absent
[6]	unsupervised	Not mentioned	To fit model to human reading time and cloze data
[66]	Both	Group depending	Blank accuracy (BA) Passage Accuracy (PA)
[67]	Not mentioned	Not mentioned	CWALS metric of Bentz et al [69]

Conclusion and future directions

Each language has its own grammar. Likewise, while predicting the word each grammar should be precise. Numbers of algorithms and metrics can be used for word prediction. But are all these algorithms good for predicting the word? On our findings dataset also plays a significant part in improving the model's performance. The model performs better with more high-quality datasets. In the research papers we have gone through most common datasets used are Penn tree bank, wordnet and Wiki-Text. As for the algorithms LSTM, RNN, CNN and cosine similarity are handed down. Ambiguous is the main concern for word prediction. Moreover, it also depends on which language we are looking to predict. Arabian, Turkish, Urdu, Chinese are extremely tough to predict than that of English. English being the globally renowned language. Quality datasets can easily be found. But for the languages like Urdu due to lack of the dataset prediction is difficult. For the bangla language researchers are still searching the best model performance. Some Algorithms for instance Stupid BackOff algorithm (SBO) is extremely handy for working on larger datasets. Auto-completion and sequence prediction can be a core component in query-related systems or recommendation-based interfaces to integrate the system with human engagement. It improves the user interaction by guessing the word(s) that the user wants to type. The terms that are offered are chosen based on their probability ranking. The predicted word is the one with the highest chance after two prior words.

21st century is the world full of messaging. While chatting with our friends and family we often send emojis. It is another way of showing your feelings. Emoji predictions are made using recurrent neural networks. The principal layer, the hidden layer, and the last layer would all be part of this methodology's repeated neuronal organisation. For this, a bigram and a trigram module can be created in python. Hashmaps can be used to perform faster lookups. The measure of unstructured text-based information generated by social collaboration among persons has transformed into an enormous secret riches of information thanks to the Internet and the approach of web-based media. So, to take advantage of such significant experiences for dynamic purposes, text-based information should be handled to remove noteworthy bits of knowledge in a machine meaningful structure by utilizing Natural Language Processing (NLP) procedures. A directed methodology called L2AWE (Learning To Adapt with Word Embeddings) targets adjusting a NER framework prepared on a source order pattern to a given objective one.

Because this is the first time prediction models for the Kurdish language have been developed for both the Sorani and Kurmanji dialects, it was more difficult to generate this corpus and these grammes than it would be for the English language. In Dzongkha, a word can be made up of a single syllable or many

syllables. 6 and 22 keystrokes are required for a single syllable and multiple syllabic word, respectively. The majority of the syllables and words take several keystrokes. Word embeddings collect both syntactic and semantic information in order to perform natural language processing tasks. Understanding natural languages requires such a complicated tree-like structure. Recurrent neural networks (RNNs) represent languages in a sequential manner and are unable to appropriately encode hierarchical syntactic relationships, resulting in underperformance on comprehension-based tasks. As a result, a novel neural language model known as relative syntactic distance LSTM (RSD-LSTM) may be used to dynamically express syntactic structural relationships. AWD-LSTM was the baseline model, which produces outcomes that are cutting-edge on the PTB and WT2 datasets. The convolution part of CNN is presented into the language model of succession expectation task to fully mine the nearby element data between word groupings and work on the model's prediction. The MCNN-ReMGU model can successfully obtain globalisation and localization highlight data between word arrangements, as well as unearth the word grouping association in sentences, hence improving the model's word expectation capacity. CNN-ReMGU outperforms standard approaches in the exhibition of the word expectation task.

Unigram-normalized perplexity (PPLu), also known as negated mutual information between a word and its context information, is a novel metric, for evaluating MSLMs with varying vocabulary sizes. PPLu values were determined by combining PPL and input text probabilities. Cloze Distillation shows how altering training incentives away from corpus probability and toward psycholinguistic errand-based modeling improves cognitive and linguistic models. In any case, there are a few potential clarifications for the impact Cloze Distillation has an effect on the performance of language models, given that a few models anticipate perusing times beyond the cloze information acquired in Provo. Cloze errand generates data that is a more reliable representation of the assumptions generated from human reading and hence ready to direct the models toward a more human-like arrangement of assumptions.
The PMI merely estimates how often a word pair will appear in a series of sentences. Internal sentence structures and the relative placement of words within particular phrases are not taken into account. Fine-tuning BERT yields the best results among other models; however, it still falls short of human performance. BERT and mBERT suggests that the latter exhibits evidence of learning in many languages, that it learns these types of rules more quickly in some languages than others, and that it has a lower sensitivity to syntax than monolingual BERT.

In mainstream spoken language processing systems, automatic speech recognition (ASR) and a specialized natural language module are frequently integrated in a cascade. A low-WER ASR module is necessary to prevent such cascaded error propagation. Another method is to prevent experimental error by utilizing a direct final model or making sure there is adequate data to train end-to-end models. Final models are simpler to construct, get less inertia, and, in some cases, perform better than cascaded systems. To achieve unrivalled ST execution, one should either avoid experimental error in cascaded systems or make sure there is enough data to train final algorithms. Consistent word depictions, also known as word embeddings, are used in both scenarios. During training, speech models convert speech signals to matched text and understand the relationships between speech and text. Speech models are created using the data and relationships communicated by word embeddings. It allows speech models to refer to additional semantics in the output text, allowing them to capture precise meaning. The sequence-to-sequence ASR models are taken into consideration, with an autoregressive decoder being used to anticipate the record relating to the information discourse. The decoder is given the task of predicting word implanting in this case. End-to-end ASR frameworks achieve execution that is almost comparable to (or astonishingly better than) normal ASR due to their varied limits and constructing discourse record sets. Word embeddings are used as a transitory in executing many operations in the end-to-end architecture, and it helps with interpretation quality. Furthermore, with the provided methodologies, a

pre-trained strategy is found to bring additional enhancements. The results suggest that learning acoustic-to-semantic transitions is possible. They are used as a transitory in executing many operations in the end-to-end architecture, and it helps with interpretation quality. Another major challenge in the future will be the case of innovative or unusual material specifications that cannot be addressed using word embeddings since they are not part of their lexicon. This is particularly important in online media, as users develop new words and abbreviations. This might be solved by merging personality and word representations in a single framework, and explicitly preparing word embeddings models using corpora would also be interesting.

References:

[1] Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., et al. (2016). Deep speech 2: End-to-end speech recognition in english and mandarin. In Proc. ICML (pp. 173–182).
[2] S. Merity, N. S. Keskar, R. Socher, Regularizing and optimizing LSTM language models, arXiv: 1708.02182 (2017).
[3] Shuang, K., Tan, Y., Cai, Z., & Sun, Y. (2020). Natural language modeling with syntactic structure dependency. Information Sciences, 523, 220-233.
[4] T. Mikolov , M. Karafiát , L. Burget , J. ˇCernock `y , S. Khudanpur , Recurrent neural network based language model, in: Eleventh Annual Conference of the International Speech Communication Association, 2010 .
[5] Nozza, D., Manchanda, P., Fersini, E., Palmonari, M., & Messina, E. (2021). LearningToAdapt with word embeddings: Domain adaptation of Named Entity Recognition systems. Information Processing & Management, 58(3), 102537.
[6] Eisape, T., Zaslavsky, N., & Levy, R. (2020, November). Cloze Distillation: Improving Neural Language Models with Human Next-Word Prediction. In CoNLL (pp. 609-619).
[7] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. OpenAI Blog, 1(8):9.
[8] Mahte, R., Nair, R., Nair, V., Pillai, A., & Kulkarni, M. (2020). Emoticon Suggestion with Word Prediction using Natural Language Processing
[9] Taichi Matsui and Shohei Kato,'Emoticon Recommendation System Reflecting User Individuality-A Preliminary Survey of Emoticon Use', ICAART, 2017
[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
[11] Roh, J., Park, S., Kim, B. K., Oh, S. H., & Lee, S. Y. (2021). Unsupervised multi-sense language models for natural language processing tasks. Neural Networks.
[12] Hasan, A. T., Mohammad, M., Mahmud, H., & Hasan, M. K. (2020, January). Mutual Context-based Word Prediction for Internet Messenger Chat. In Proceedings of the International Conference on Computing Advancements (pp. 1-6).
[13] Wangchuk, K., Riyamongkol, P., & Waranusast, R. (2021). Next Syllables Prediction System in Dzongkha Using Long Short-Term Memory. Journal of King Saud University-Computer and Information Sciences.
[14] M. Marcus, B. Santorini, M.A. Marcinkiewicz, Building a large annotated corpus of english: the penn treebank (1993).
[15] S. Merity, C. Xiong, J. Bradbury, R. Socher, Pointer sentinel mixture models, arXiv: 1609.07843 (2016)
[16] Miller, G. A., Leacock, C., Tengi, R., & Bunker, R. T. (1993). A semantic concordance. In Proc. of the workshop on human language technology (pp. 303–308).Association for Computational Linguistics.

[17] Marcus, M. P., Marcinkiewicz, M. A., & Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. Computational Linguistics, 19(2), 313–330.
[18] Mikolov, T., Joulin, A., Chopra, S., Mathieu, M., & Ranzato, M. (2014). Learning longer memory in recurrent neural networks. ArXiv preprint.
[19] Rakib, O. F., Akter, S., Khan, M. A., Das, A. K., & Habibullah, K. M. (2019, December). Bangla word prediction and sentence completion using GRU: an extended version of RNN on N-gram language model. In 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI) (pp. 1-6). IEEE.
[20] Kumhar, S. H., Kirmani, M. M., Sheetlani, J., & Hassan, M. (2021). Word Embedding Generation for Urdu Language using Word2vec model. *Materials Today: Proceedings*.
[21] Babu, A. S., & Kumar, P. N. V. S. P. (2010). Comparing neural network approach with n-gram approach for text categorization. International Journal on Computer Science and Engineering, 2(1), 80-83.
[22] Dineshika Dulanjalee Wijerathna, 'Emoticon Suggestion based on Recurrent Neural Network', University of Moratuwa, 2017.
[23] Ruobing Xie1, Zhiyuan Liu1 , Rui Yan2 and Maosong Sun1,'Neural Emoji Recommendation in Dialogue Systems',2016
[24] Jaysidh Dumbali, Nagaraja Rao A. ,'Real Time Word Prediction Using N-Grams Model', International Journal of Innovative Technology and Exploring Engineering (IJITEE) , Volume8,Issue-5, March 2019
[25] G. Tur and R. De Mori, Spoken Language Understanding: Systems for Extracting Semantic Information from Speech. Hoboken, NJ, USA: Wiley 2011.
[26] M. Sperber, G. Neubig, J. Niehues, and A. Waibel, "Attention-passing models for robust and data-efficient end-to-end speech translation," Trans. Assoc. Comput. Linguist., vol. 7, pp. 313–325, 2019.
[27] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Adv. Neural Inform. Process. Syst., 2013, pp. 3111–3119.
[28] Y. Bengio, P. Simard, and P. Frasconi, ''Learning long-term dependencies with gradient descent is difficult,'' IEEE Trans. Neural Netw., vol. 5, no. 2, pp. 157–166, Mar. 1994.
[29] G.-B. Zhou, J. Wu, C.-L. Zhang, and Z.-H. Zhou, ''Minimal gated unit for recurrent neural networks,'' Int. J. Autom. Comput., vol. 13, no. 3, pp. 226–234, Jun. 2016.
[30] T. Mikolov and G. Zweig, ''Context dependent recurrent neural network language model,'' in Proc. IEEE Spoken Lang. Technol. Workshop (SLT), Miami, FL, USA, Dec. 2012, pp. 234–239
[31] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[32] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5–8, 2013, Lake Tahoe, Nevada, United States, 2013, pp. 3111–3119.
[33] A. Mnih, K. Kavukcuoglu, Learning word embeddings efficiently with noisecontrastive estimation, in: Advances in Neural Information Processing Systems 2013. Proceedings of a meeting held December 5–8, 2013, Lake Tahoe, Nevada, United States, 2013, pp. 2265–2273.
[34] Lafferty, J. D., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the 18th international conference on machine learning (pp. 282–289)
[35] Rizzo, G., & Troncy, R. (2012). NERD: A framework for unifying named entity recognition and disambiguation extraction tools. In Proceedings of the Demonstrations at the 13th conference of the European chapter of the association for computational linguistics (pp. 73–76)

[36] Ramage, D., Hall, D., Nallapati, R., & Manning, C. D. (2009). Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In Proceedings of the 2009 conference on empirical methods in natural language processing (pp. 248–256).
[37] T. Ruas, C. H. P. Ferreira, W. Grosky, F. O. de França, and D. M. R. de Medeiros, ''Enhanced word embeddings using multi-semantic representation through lexical chains,'' Inf. Sci., vol. 532, pp. 16–32, Sep. 2020
[38] H. P. Shin, M. H. Kim, Y. M. Jo, H. Y. Jang, and A. Cattle, ''Annotation Scheme for Constructing Sentiment Corpus in Korean,'' in Proc. 26th Pacific Asia Conf. Lang., Inf. Comput., 2012, pp. 181–190.
[39] O. V. Berkout, A. J. Cathey, and K. K. Kellum, ''Scaling-up assessment from a contextual behavioral science perspective: Potential uses of technology for analysis of unstructured text data,'' J. Contextual Behav. Sci., vol. 12, pp. 216–224, Apr. 2019.
[40] Sennrich, R., Haddow, B., & Birch, A. (2015). Improving neural machine translation models with monolingual data. arXiv preprint arXiv:1511.06709.
[41] Luke, S. G., & Christianson, K. (2018). The Provo Corpus: A large eye-tracking corpus with predictability norms. Behavior research methods, 50(2), 826-833.
[42] Luke, S. G., & Christianson, K. (2018). The Provo Corpus: A large eye-tracking corpus with predictability norms. Behavior research methods, 50(2), 826-833.
[43] Hill, J., & Simha, R. (2016, June). Automatic generation of context-based fill-in-the-blank exercises using co-occurrence likelihoods and Google n-grams. In Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications (pp. 23-30).
[44] Zweig, G., & Burges, C. J. (2011). The microsoft research sentence completion challenge. Microsoft Research, Redmond, WA, USA, Technical Report MSRTR-2011, 129.
[45] Mostafazadeh, N., Chambers, N., He, X., Parikh, D., Batra, D., Vanderwende, L., ... & Allen, J. (2016, June). A corpus and cloze evaluation for deeper understanding of commonsense stories. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 839-849).
[46] Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1192–1202, Brussels, Belgium. Association for Computational Linguistics.
[47] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
[48] Marten van Schijndel, Aaron Mueller, and Tal Linzen. 2019. Quantity doesn't buy quality syntax with neural language models. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5835–5841, Hong Kong, China. Association for Computational Linguistics.
[49] Mandic, D., & Chambers, J. (2001). Recurrent neural networks for prediction: learning algorithms, architectures and stability. Wiley.
[50] Shen, Y., Lin, Z., Jacob, A. P., Sordoni, A., Courville, A., & Bengio, Y. (2018). Straight to the tree: Constituency parsing with neural syntactic distance. arXiv preprint arXiv:1806.04168.
[51] Gogineni, A. K., Swayamjyoti, S., Sahoo, D., Sahu, K. K., & Kishore, R. (2020). Multi-Class classification of vulnerabilities in Smart Contracts using AWD-LSTM, with pre-trained encoder inspired from natural language processing. IOP SciNotes, 1(3), 035002.
[52] Ardoin, S. P., Martens, B. K., & Wolfe, L. A. (1999). Using high-probability instruction sequences with fading to increase student compliance during transitions. Journal of Applied Behavior Analysis, 32(3), 339-351.

[53] Vajda, S., & Santosh, K. C. (2016, December). A fast k-nearest neighbor classifier using unsupervised clustering. In International conference on recent trends in image processing and pattern recognition (pp. 185-193). Springer, Singapore.
[54] Roh, J., Park, S., Kim, B. K., Oh, S. H., & Lee, S. Y. (2021). Unsupervised multi-sense language models for natural language processing tasks. Neural Networks.
[55] Hasan, A. S. M., & Mohammad, M. (2018). Mutual Context Based Word Prediction (Doctoral dissertation, Department of Computer Science and Engineering, Islamic University of Technology, Board Bazar, Gazipur, Bangladesh).
[56] Zobaed, S., Haque, M. E., Rabby, M. F., & Salehi, M. A. (2021, January). Senspick: sense picking for word sense disambiguation. In 2021 IEEE 15th International Conference on Semantic Computing (ICSC) (pp. 318-324). IEEE.
[57] Valentine, H. T., Gregoire, T. G., & Furnival, G. M. (1987). Unbiased estimation of total tree weight by three stage sampling with probability proportional to size. of: Wharton, EH, & Cunia, T.(eds), Tree Biomass Regression Functions and their Contribution to the Error of Forest Inventory Estimates. General Technical Report NE-GTR-117. Broomall, Pennsylvania: USDA Forest Service, 129-132.
[58] Barman, P. P., & Boruah, A. (2018). A RNN based Approach for next word prediction in Assamese Phonetic Transcription. Procedia computer science, 143, 117-123.
[59] Jamtsho, Y., & Muneesawang, P. (2020). Dzongkha Word Segmentation using Deep Learning. In 2020 12th International Conference on Knowledge and Smart Technology (KST) (pp. 1-5). IEEE.
[60] Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with ReLU activation function. The Annals of Statistics, 48(4), 1875-1897.
[61] Ahmadi, S. (2021). Hunspell for Sorani Kurdish Spell Checking and Morphological Analysis. arXiv preprint arXiv:2109.06374.
[62] Veisi, H., MohammadAmini, M., & Hosseini, H. (2020). Toward Kurdish language processing: Experiments in collecting and processing the AsoSoft text corpus. Digital Scholarship in the Humanities, 35(1), 176-193.
[63] Niesler, T. R., & Woodland, P. C. (1996, May). A variable-length category-based n-gram language model. In 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings (Vol. 1, pp. 164-167). IEEE.
[64] Atlınar, F., Ayar, T., Darrige, A., AlQays, S., Bağcı, A., & Amasyali, M. F. (2020, October). Masked Word Prediction with Statistical and Neural Language Models. In 2020 Innovations in Intelligent Systems and Applications Conference (ASYU) (pp. 1-4). IEEE.
[65] Fourtassi, A., Börschinger, B., Johnson, M., & Dupoux, E. (2013, August). Why is english so easy to segment?. In Proceedings of the fourth annual workshop on cognitive modeling and computational linguistics (cmcl) (pp. 1-10).