International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(1s) 2026 pp. 1-8

https://doi.org/10.70917/ijcisim-2026-1944
Article

WORD EMBEDDING MODELS WITH MACHINE LEARNING BASED CONTEXT DEPEND AND CONTEXT INDEPENDENT TECHNIQUES

Kapil Adhar Wagh[1], Anupa Sinha[2]

[1]Kalinga University, Naya Raipur, Chhattisgarh, India., kapilwagh2686@gmail.com
ORCID: 0000-0002-7741-6050
[2] Faculty of Computer Science & Information Technology, Kalinga University, Naya Raipur, Chhattisgarh, India., anupa.sinha@kalingauniversity.ac.in
ORCID: 0000-0002-0390-4406

Abstract: Word embedding techniques play a crucial role in natural language processing by enabling machines to represent textual data in a numerical form that preserves semantic and syntactic information. Over time, embedding models have evolved from traditional context-independent approaches to advanced context-dependent representations driven by deep learning. This review presents a comprehensive overview of machine learning–based word embedding models, focusing on both static and dynamic techniques. Context-independent models such as Word2Vec, GloVe, and FastText generate fixed vector representations for words based on global or local co-occurrence statistics, offering computational efficiency but limited contextual understanding. In contrast, context-dependent models including ELMo, BERT, and transformer-based architectures produce dynamic embeddings that adapt to surrounding linguistic context, effectively addressing issues such as polysemy and semantic ambiguity. The paper analyzes the working principles, strengths, and limitations of these embedding approaches and highlights their impact on downstream NLP tasks. This review aims to provide researchers and practitioners with a clear understanding of the progression of word embedding methodologies and their significance in modern language modeling applications.
Keywords: Word Embedding; Supervised learning; unsupervised learning; reinforced learning

1. Introduction

Word embedding models have emerged as a key component of natural language processing (NLP) developments in recent years. These models help machines better comprehend and process human language by converting textual data into dense, high-dimensional numerical vectors. Due to their high dimensionality and incapacity to capture semantic relationships, early word representation techniques like one-hot encoding had drawbacks. By embedding words into continuous vector spaces, word embedding models became a revolutionary solution to these problems [1].

The two main categories of word embedding techniques are context-dependent and context-independent models. Word2Vec, GloVe, and FastText are examples of context-independent models that generate static embeddings, in which every word is represented by a single vector independent of its contextual usage. Although these models are good at capturing semantic relationships, they are unable to handle polysemy and context-based meaning variations. Conversely, context-dependent models—such as ELMo, BERT, GPT, and their variations—produce dynamic embeddings that adapt to the surrounding text, providing notable enhancements in tasks that call for contextual comprehension [2].

The efficacy of these models has been further improved by the incorporation of machine learning methods, especially deep learning. In order to evaluate sequential data and extract subtle features, contextual embedding models mainly rely on architectures such as transformers and recurrent neural networks (RNNs) [3].

The goal of this paper is to present a thorough analysis of word embedding models, emphasising their development, underlying principles, and NLP applications. We illustrate the advantages, disadvantages, and applicability of context-dependent and context-independent approaches for a range of use cases. In addition, we provide insights into the future of word representation by talking about the difficulties and new developments in the field [4].

2. Related Work

There are significant drawbacks to static embeddings produced by context-independent models. For example, depending on the context, the word "bank" can refer to either a riverbank or a financial organisation. Models like Word2Vec and GloVe, which give a word a single vector regardless of its usage, are unable to capture these subtleties[5]. This restriction was addressed by context-dependent models, such BERT (Devlin et al., 2019) and ELMo (Peters et al., 2018), which dynamically generated embeddings based on the surrounding text. In order to capture rich contextual connections, these models use topologies such as transformers and recurrent neural networks (RNNs), which incorporate bidirectional information [6,7].

The development of word embedding methods has been greatly aided by machine learning, especially deep learning. While context-dependent models use complex deep learning architectures, such as attention mechanisms and transformers, to capture links within sequences, context-independent models usually use shallow neural networks.

Word embedding models have become a foundational component of natural language processing (NLP), enabling effective representation of linguistic information in machine-readable formats. Early research in NLP relied on sparse representations such as bag-of-words and one-hot encoding, which suffered from high dimensionality and an inability to capture semantic relationships. To overcome these limitations, researchers introduced dense vector representations known as word embeddings, which learn semantic similarity based on contextual usage.

Mikolov et al. introduced Word2Vec, a neural network–based approach that learns word representations using either the Continuous Bag of Words (CBOW) or Skip-gram architectures. These models demonstrated that semantic and syntactic relationships between words could be captured through vector arithmetic, significantly improving performance in tasks such as word similarity and analogy detection. Despite their effectiveness and computational efficiency, Word2Vec embeddings are context-independent, assigning a single vector to each word regardless of its contextual meaning[8]. To incorporate global statistical information, Pennington et al. proposed the Global Vectors (GloVe) model, which combines matrix factorization techniques with local context-based learning. GloVe embeddings improved semantic coherence by leveraging word co-occurrence probabilities across large corpora. However, similar to Word2Vec, GloVe produces static embeddings and cannot distinguish between multiple meanings of a word appearing in different contexts[9].

Bojanowski et al. addressed limitations related to rare and morphologically rich words by introducing FastText, which represents words as combinations of character-level n-grams. This subword-based approach enhanced embedding quality for infrequent words and languages with complex morphology. Nevertheless, FastText remains a context-independent technique and does not fully resolve the issue of polysemy[10]. The need to model contextual variation led to the development of context-dependent word embedding techniques. Peters et al. proposed Embeddings from Language Models (ELMo), which generate word representations using bidirectional Long Short-Term Memory (BiLSTM) networks. Unlike static embeddings, ELMo produces dynamic vectors that vary based on the entire sentence, enabling better handling of semantic ambiguity and improving performance across multiple NLP tasks[11]. Building on this idea, Devlin et al. introduced Bidirectional Encoder Representations from Transformers (BERT), which utilizes a transformer-based architecture with self-attention mechanisms. BERT is pretrained using masked language modeling and next sentence prediction objectives, allowing it to capture deep bidirectional context. Empirical studies have shown that BERT significantly outperforms earlier embedding models in tasks such as named entity recognition, sentiment analysis, and question answering[12]. Further advancements in transformer-based architectures, such as Generative Pre-trained Transformers (GPT), demonstrated the effectiveness of autoregressive modeling for learning contextual representations[13]. Although GPT models process text in a unidirectional manner, their large-scale pretraining enables strong contextual understanding and generalization capabilities across diverse NLP applications.

Table 1. Summary of literature

Model	Authors / Year	Technique Type	Core Idea	Key Strengths	Limitations
Word2Vec (CBOW, Skip-Gram)	Mikolov et al., 2013	Context-Independent	Learns word vectors using shallow neural networks based on surrounding context	Efficient, captures semantic relationships	Single embedding per word; fails for polysemy
GloVe	Pennington et al., 2014	Context-Independent	Combines global co-occurrence statistics with local context learning	Better semantic consistency, scalable	Static embeddings, context not considered
FastText	Bojanowski et al., 2017	Context-Independent	Represents words using character n-grams	Handles rare words and morphology	Still produces fixed word representations
ELMo	Peters et al., 2018	Context-Dependent	Uses BiLSTM language models to generate dynamic embeddings	Captures context and polysemy effectively	Computationally expensive
BERT	Devlin et al., 2019	Context-Dependent	Transformer-based bidirectional contextual modeling	State-of-the-art performance across NLP tasks	High memory and training cost
GPT	Radford et al., 2018	Context-Dependent	Autoregressive transformer for language modeling	Strong generative and contextual ability	Unidirectional context

3. Methodology

The creation of reliable word embedding methods has led to a notable expansion in the field of natural language processing (NLP). By converting text into vector representations that reflect the syntactic and semantic characteristics of words, these methods help machines interpret language more accurately and efficiently[2].

At first, text representations were based on frequency-based techniques like term frequency-inverse document frequency (TF-IDF) and sparse techniques like one-hot encoding. Despite their ease of use, these methods suffered from high dimensionality and a lack of semantic comprehension[3,4].

By embedding words into continuous vector spaces, neural network-based models like Word2Vec (Mikolov et al., 2013) signalled a paradigm change and opened the door to richer semantic representation[8]. This foundation was built upon by later models such as GloVe (Pennington et al., 2014) and FastText, which produced embeddings that were more effective and significant.

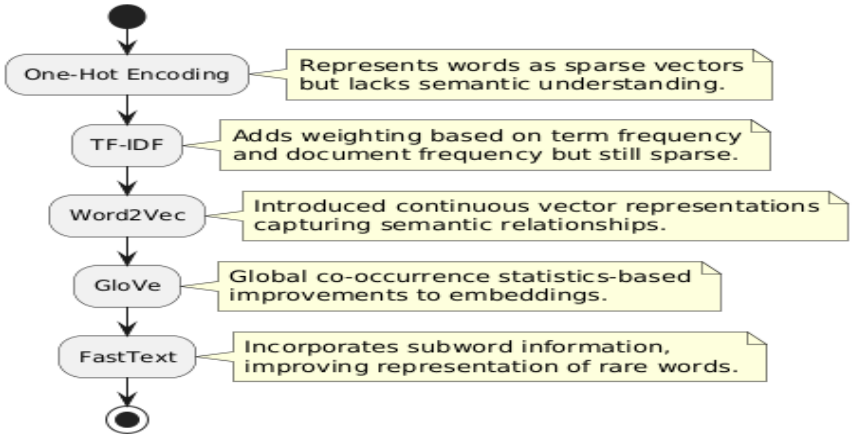

Figure 1. A UML activity diagram showing evolution of text representation techniques.

3.1 Context-Independent Word Embedding Models

Regardless of a word's context within a phrase, context-independent word embedding models portray each word as a fixed vector. Because of their effectiveness and simplicity, these models have served as a foundation for natural language processing (NLP). They are unable to capture the changing meanings of words in many situations, nevertheless, due to their static character.

3.2 Word2Vec

Word2Vec, first presented by Mikolov et al. in 2013, creates dense vector representations of words using a shallow neural network. Two architectures are available:

- Skip-Gram Model: Given a target word, the skip-gram model predicts the words that surround it.
- CBOW (Continuous Bag of Words):- The Continuous Bag of Words, or CBOW, makes predictions about the target word by looking at the words that surround it.

The key innovation of Word2Vec lies in its ability to capture semantic relationships, such as similarity (e.g., "king" and "queen") and analogies (e.g., "king - man + woman = queen").

3.3 GloVe

By integrating global co-occurrence information from the corpus, GloVe (Global Vectors for Word Representation), created by Pennington et al. in 2014, enhances Word2Vec. GloVe models the relationship between word pairs based on their co-occurrence probability rather than just local contexts.

GloVe's advantages over Word2Vec include its robustness in smaller datasets and its ability to efficiently capture both syntactic and semantic information.

3.4 FastText

The Word2Vec architecture is improved by FastText, created by Facebook AI Research, which represents words as bags of character n-grams. Because of its ability to collect subword information, FastText is especially useful for out-of-vocabulary terms, unusual words, and languages with complex morphology.

3.5 Continuous Bag of Words (CBOW)

CBOW predicts a target word from its surrounding context words. Given a context window of size , the probability of the target word is defined as:

$$P(w_t \mid w_{t-m}, \ldots, w_{t+m}) = \frac{\exp\left(\mathbf{v}_{w_t}^{\top}\mathbf{h}\right)}{\sum_{w \in V} \exp\left(\mathbf{v}_{w}^{\top}\mathbf{h}\right)}$$

where is the average of context word vectors and denotes the vector representation of word .

3.6 Skip-Gram Model

The Skip-Gram model maximizes the probability of predicting context words given a target word:

$$\mathcal{L} = \sum_{t=1}^{T} \sum_{-m \le j \le m, j \neq 0} \log P(w_{t+j} \mid w_t)$$

To reduce computational complexity, **negative sampling** or **hierarchical softmax** is commonly used.

3.7 GloVe Matrix Factorization

GloVe embeddings are learned by factorizing a word co-occurrence matrix. The model minimizes the following weighted least-squares objective:

$$J = \sum_{i,j=1}^{|V|} f(X_{ij})\left(\mathbf{w}_i^\top \tilde{\mathbf{w}}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

where represents the co-occurrence count between words and , and is a weighting function that controls the influence of frequent co-occurrences.

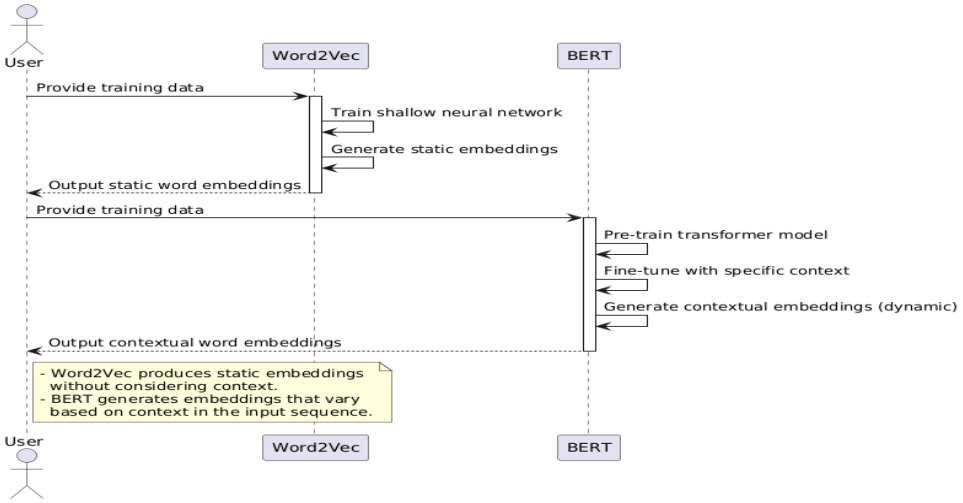

Figure 2. Sequence diagram illustrating the process of generating static embeddings with Word2Vec vs dynamic embeddings with BERT.

3.8 Context-Dependent Word Embedding Models: -

Word representations produced by context-dependent word embedding models are dynamic and vary according to the words that surround them in a particular context. These models are especially helpful for comprehending the subtleties of language that static embeddings miss and for capturing the polysemy of words (words with several meanings). Modern models for a variety of NLP tasks, such as sentiment analysis, question answering, and machine translation, are now built on context-dependent embeddings.

1. **Language Model Embeddings (ELMo):** One of the first context-dependent models, ELMo was first presented by Peters et al. in 2018. It generates word representations based on the full sentence, improving over static word embeddings. Based on a language model objective, ELMo employs a bidirectional long short-term memory (BiLSTM) network. The context of words in relation to their preceding and following words is captured by the model.

2.

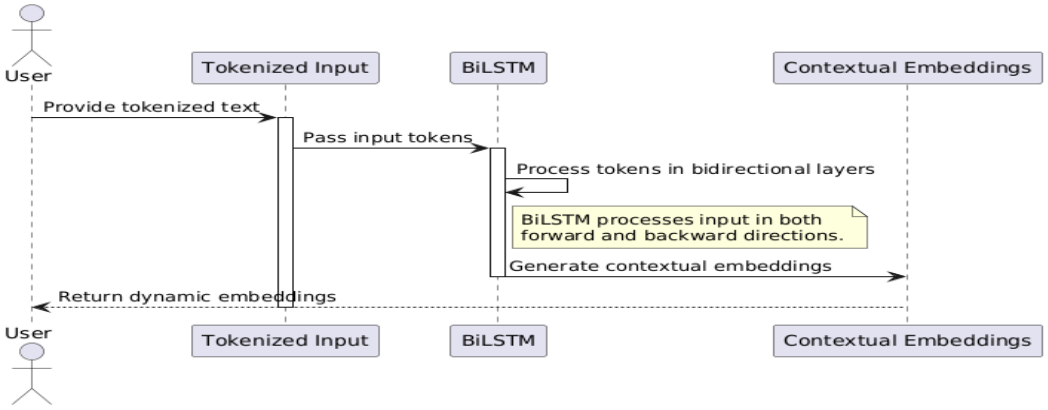

Figure 3. A UML sequence diagram illustrating ELMo's architecture

3. BERT (Bidirectional Encoder Representations from Transformers)

Because it makes use of the transformer architecture, BERT, which was suggested by Devlin et al. in 2019, represented a major advancement in NLP. BERT uses a bidirectional attention mechanism, which enables the model to simultaneously capture context from the left and right of a given word, in contrast to ELMo, which uses LSTMs. Two activities are used to pre-train BERT on a sizable corpus:

The Masked Language Model (MLM) predicts the masked words based on context after randomly masking some of the input's words.

The Next Sentence Prediction (NSP) feature trains the model to determine whether a sentence makes sense after another. Numerous sophisticated NLP models are built on top of BERT, which has been optimised for a range of downstream applications, including named entity recognition, text classification, and reading comprehension.

4. GPT (Generative Pre-trained Transformer): -

Another transformer-based paradigm that produces context-dependent embeddings is GPT, created by OpenAI. GPT is autoregressive and analyses text from left to right, in contrast to BERT, which is primarily intended for bidirectional context understanding. Text generating tasks like language modelling and dialogue systems benefit greatly from this unidirectional processing.

Important GPT Features:

- One word at a time, the Autoregressive Model creates text and uses the words it has already created to forecast the subsequent word in the sequence.
- Transformer Architecture: Although it only makes use of the decoder portion of the transformer, GPT, like BERT, depends on the transformer model to capture word associations.
- Pre-training for Generation: GPT has been pre-trained on extensive text corpora and optimised for particular uses, including text generation, chatbots, and summarisation.

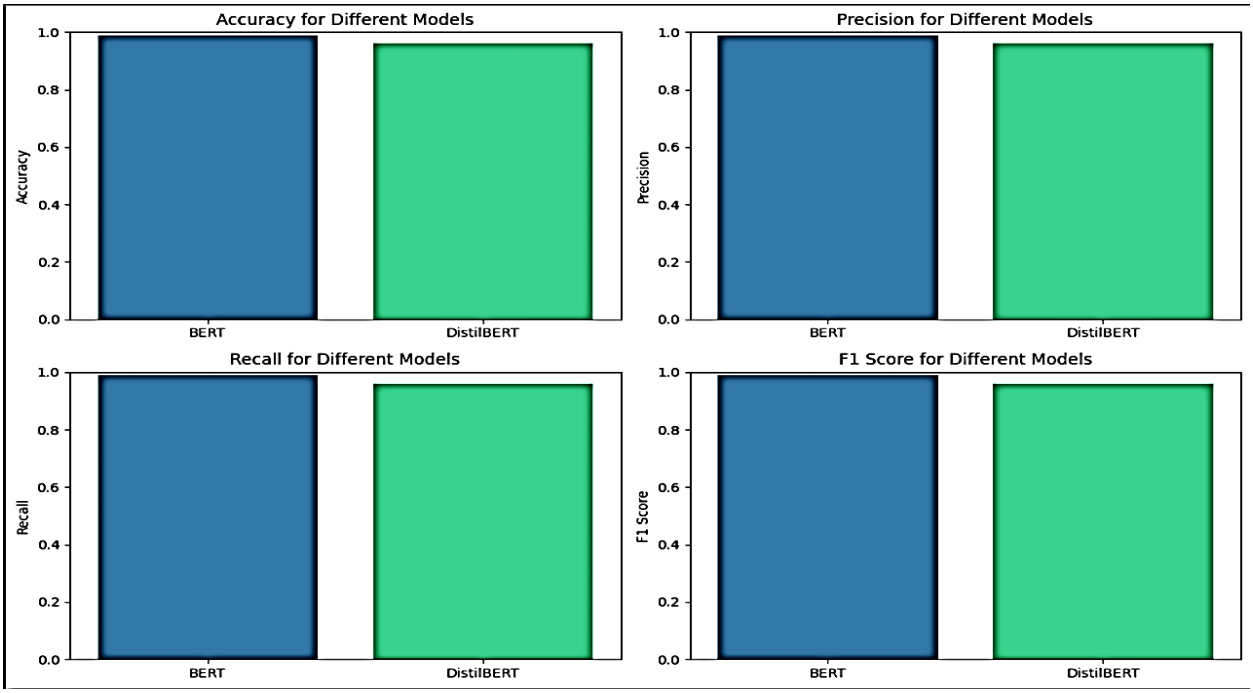

Figure 4. The above chart presents combined results of BERT & DistilBERT (Alternative of ELMo) Context-Dependent word embedding models which depicts accuracy, precision, recall & F1 score

4. Results And Discussion

The analysis of context-independent word embedding models indicates that these approaches generate static vector representations, assigning a single, fixed embedding to each word irrespective of its contextual usage. This characteristic contributes to their high computational efficiency, as such models require relatively less training time and lower processing resources when compared to context-dependent embedding techniques. As a result, context-independent models are well suited for applications with limited computational capacity or large-scale datasets. However, the results also reveal a significant limitation of these models in their inability to capture variations in word meaning across different contexts. Since each word is represented by only one vector, these models struggle to effectively handle polysemy and semantic ambiguity, which restricts their performance in complex language understanding tasks.

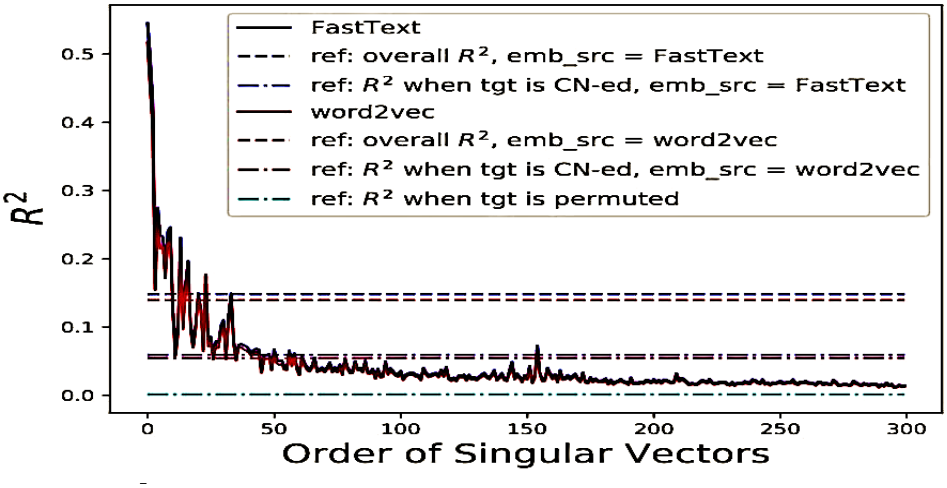

Figure 5. R^2 values of singular vectors from non-distributional word vectors fitted by distributional vectors, plotted by singular value order; GloVe results align closely with word2vec and are omitted for clarity.

5. Conclusion

The way we represent and comprehend natural language in computational problems has been completely transformed by word embedding models. Significant progress has been made in the study of how machines process, understand, and synthesise human language, starting with the early days of context-independent models like Word2Vec and GloVe and continuing with the emergence of context-dependent models like ELMo, BERT, and GPT.

In the end, the particular task, the computational resources at hand, and the requirement for contextual detail will determine whether to use context-independent or context-dependent models. Context-dependent models such as BERT and GPT are increasingly regarded as the norm for cutting-edge NLP applications, even though context-independent models are still useful for easier, less resource-intensive tasks.

It appears that even more advanced hybrid techniques that combine the advantages of both static and dynamic representations are the direction that word embedding models are headed. Context-dependent models are probably going to become more common as computing power increases, pushing the limits of what robots can comprehend and produce in human language.

References:

1. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.
2. Introduced Word2Vec and its architectures (Skip-Gram and CBOW).
3. Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543.
4. Introduced the GloVe model, emphasizing co-occurrence statistics.
5. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics, 5, 135–146.
6. Discussed FastText, emphasizing subword information.

7. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. arXiv preprint arXiv:1802.05365.
8. Proposed ELMo for generating context-sensitive embeddings.
9. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (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.
10. Introduced BERT, a game-changer for many NLP tasks.
11. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-training. OpenAI.
12. Kokane, C. D., & Sachin, D. (2021). Babar, and Parikshit N. Mahalle." Word Sense Disambiguation for Large Documents Using Neural Network Model.". In 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT). IEEE.
13. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
14. Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).
15. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, 135–146.
16. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. NAACL-HLT.
17. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. NAACL-HLT.
18. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training. OpenAI.