


A Review of Contextualized Word Embeddings and Pre-Trained Language Models, with a Focus on GPT and BERT

Maya Thapa¹, Puneet Kapoor¹, Sakshi Kaushal² and Ishani Sharma¹

¹*Yogananda School of AI, Computers and Data Sciences, Shoolini University, Solan 173212, India*

²*University Institute of Engineering and Technology, Panjab University, Chandigarh 160014, India*


Keywords: Word Embeddings, Vectors, ELMo, GPT, BERT.

Abstract: Word meanings are attempted to be encapsulated by word embeddings, which are n-dimensional distributed representations of text. Multiple computational layers are used by deep learning models to obtain hierarchical data representations. Word embedding as a deep learning approach has attracted a lot of attention and is used in many Natural Language Processing (NLP) applications such as text classification, sentiment analysis, named entity recognition, and topic modeling. Thus, adopting suitable deep-learning models and word embeddings is essential for getting greater results. Highlighting the substantial impact of word embeddings on NLP over the past decade, the study transitions into a detailed examination of contextualized embeddings. It addresses the limitations of conventional static embeddings and introduces the revolutionary nature of contextualized embeddings, particularly in capturing both syntactic and semantic subtleties. The paper serves as a comprehensive review of pre-trained language models, emphasizing advancements in NLP applications. Models like GPT-3 and BERT take center stage in the analysis, showcasing their strengths and limitations.

1 INTRODUCTION

In the current automated environment, computer models are trained extensively to perform well in various areas, which has resulted in advances in network signal processing, image processing, text analysis, and video interpretation. The emphasis on textual data is still prevalent despite these developments, leading to the development of Natural Language Processing (NLP). The significance of NLP stems from its ability to fully utilize textual data and overcome obstacles to comprehend, process, and extract meaning from natural language (Asudani et al., 2023). However, texts provide a special difficulty when using NLP models. Texts are by nature non-numeric, therefore machine learning models require a transformation to make them compatible with their intrinsic preference for numerical inputs. It becomes important to represent text in numerical representations, and different techniques have been used to do this, including embeddings, count-based representation, and one-hot encoding (Patil et al., 2023). Despite being widely utilized, the earlier methods had drawbacks, such as the inability to

capture word relationships, computational inefficiency, lack of context awareness, difficulty with unseen words, and sparse outputs that occupy space inefficiently. Here's where embeddings come into play, helping to overcome these drawbacks and provide a better way to express textual data. A word's meaning can be represented as an embedding, which is a real-valued vector that depends on the surrounding text (Mars, 2022). Every word in a lexicon has a vector space position and matching numerical value. Word Embeddings strive to capture words as dense vectors, ensuring that words with similar meanings are situated nearby within the embedding space. Since strongly linked words are stored with similar representations in the context of word embeddings. For this reason, word embeddings are beneficial for a range of NLP activities, involving text classification, document clustering, and sentiment analysis (Neelima and Mehrotra, 2023). Word embeddings have changed NLP at a substantial pace over the last ten years. Contextualized embeddings, on the other hand, have proven to be revolutionary since they capture both syntactic and semantic subtleties, in contrast with regular embeddings. The present study

 <https://orcid.org/0009-0001-2711-2569>

explores the shortcomings of some of the conventional or static word embeddings.

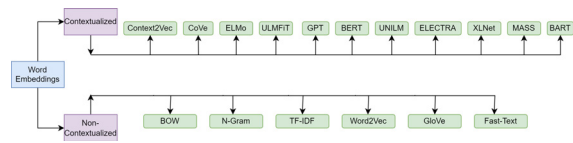


Figure 1: Various Word Embedding Techniques.

Additionally, it navigates the transition from conventional to contextual techniques, elucidating various methodologies employed in this paradigm shift as shown in Figure 1. In Section II, the paper delves into several non-contextualized or traditional word embedding techniques, and in Section III, it further explores various contextualized word embeddings.

2 NON-CONTEXTUALIZED EMBEDDINGS

Non-contextualized embeddings comprise two primary methods: frequency analysis, which looks at the entire document to identify the prominence of uncommon words by calculating their frequency and looking at co-occurrences (Bag-of-Words (Harris, 1954), N-grams (Katz, 1987), and TF-IDF (Salton et al., 1975) are examples of this type of embedding); and prediction-based, which gives words probabilities and associates each word with a vector. The latter method uses lookup tables to train static embeddings, which are represented by models such as Word2Vec (Mikolov, 2013), GloVe (Pennington et al., 2014), and FastText (Bojanowski et al., 2017). These lookup tables then turn words into dense vectors also known as singular vectors because each word is given a fixed vector representation independent of its context. The fact that word meaning might vary depending on the language or extralinguistic context is not taken into consideration by these static embeddings. Once learned, these embeddings preserve a constant context, with identical embedding tables across sentences.

3 CONTEXTUALIZED EMBEDDINGS

Contextualized word embeddings were developed to address the shortcomings of Non-Contextualized embeddings. With the use of these embeddings,

which represent words as vectors that vary depending on the linguistic environment, complex features of word meaning, such as polysemy, can be represented more thoroughly. (Hofmann et al., 2020) NLP makes extensive use of contextualized word embeddings, which form the semantic foundation of pre-trained language models that will be covered in the following sections.

3.1 Context2Vec

(Melamud et al., 2016) presented a model to learn a task-independent representation of an extensive sentential context surrounding a given target word using bidirectional long short-term memory (LSTM). Variable-length contexts can be expressed by this approach using a fixed-size vector. It utilizes a more robust parametric model based on bidirectional LSTM (Graves and Schmidhuber, 2005) instead of sticking with a basic neural model inspired by the Continuous Bag-of-words (CBOW) architecture. One LSTM processing input from left to right and another from right to left are examples of the bidirectional nature. It is noteworthy that both networks function independently and cover different sets of word embeddings from left to right and from right to left. The two LSTMs' results are combined as shown in Eq 1 (Melamud et al., 2016).

$$\text{biLS}(w_{1:n}, i) = \text{LS}(l_{1:i-1}) \oplus \text{rLS}(r_{n:i+1}) \quad (1)$$

This model views the entire sentence as generating sentential context, as opposed to CBOW's limited context window size. This all-encompassing strategy works well for capturing pertinent information even when it is far or distant from the specified target word. Consequently, words linked to similar sentential contexts exhibit comparable embeddings. A simple application like lexical substitution, (Ashihara et al., 2019) employs Context2Vec with Dependency-based Multi-Sense Embedding (DMSE) (Ashihara et al., 2018) with a viewpoint for paraphrasing the target word within a provided sentence. The evaluation findings show that this technique surpasses the state-of-the-art paraphrase strategy and is superior in lexical substitution tasks. (Hashempour and Villavicencio, 2020) used Context2Vec, which demonstrated that Context2Vec can distinguish between literal and idiomatic senses in different regions of semantic space using dimensionality reduction and lexical replacement.

3.2 Context-Vectors (CoVe)

Presented in (McCann et al., 2017), CoVe generates contextualized representations of words by using a deep LSTM encoder (Wang et al., 2016) (Shi et al., 2016) that has been trained on a machine translation (MT) job. Originally trained on MT, the attentional seq-to-seq model exhibits good transferability to other NLP tasks. The paper emphasizes that MT data serves as a reusable model foundation and shows how MT-LSTM is equivalent to ImageNet-CNN in computer vision as shown in Eq 2 (McCann et al., 2017).

$$\text{CoVe}(w) = \text{MT} - \text{LSTM}(\text{GloVe}(w)) \quad (2)$$

Adding CoVe to word vectors outperforms monolingual encoders such as language modeling in downstream task performance, particularly in semantic similarity tests. Performance on downstream tasks is positively correlated with the amount of MT-LSTM training data. For tasks like question answering and classification, the GloVe embeddings when paired with CoVe outperform models that only use GloVe, with further improvement coming from character n-gram embeddings as shown in Eq 3 (McCann et al., 2017). This highlights how CoVe is complimentary, connecting character-level insights with word-level information from GloVe to improve model performance.

$$\tilde{w} = [\text{GloVe}(w); \text{CoVe}(w)] \quad (3)$$

For the Argument Reasoning Comprehension task of SemEval 2018 (Kim et al., 2018) presented a novel neural structure comprising of three parts that together evaluate a collection of given phrases (a claim, a rationale, and a warrant) for logical coherence and plausibility. To mitigate data scarcity, the model uses transfer learning by using contextualized word vectors pretrained on large machine translation (MT) datasets. The results of quantitative analysis show that LSTMs trained on MT datasets alone outperform non-transferred models and a number of baselines, achieving an accuracy of around 70% on the development set and 60% on the test set.

3.3 Embeddings From Language Models (ELMo)

ELMo, described in (Peters et al., 2018), generates word embeddings that encapsulate semantic and

syntactic information through the utilization of a bidirectional LSTM model. ELMo's representations, in contrast to conventional word embeddings, are context-dependent and are obtained from the complete input sentence using a bidirectional language model. In a broader sense, to calculate a task specific weighting across all biLM layers it uses Eq 4 (Peters et al., 2018).

$$\text{ELMo}_k^{\text{task}} = E(R_k; \Theta^{\text{task}}) = \gamma^{\text{task}} \sum_{j=0}^L s_j^{\text{task}} h_{k,j}^{\text{LM}} \quad (4)$$

These embeddings perform better than those based only on the top LSTM layer since they are computed from all internal layers. ELMo's bidirectional LSTM captures context-dependent and semantic characteristics in higher-level states, while syntactic aspects are captured in lower-level states. The technique effectively models polysemy and significantly raises the bar for state-of-the-art performance in six NLP tasks, such as sentiment analysis and question answering. Notably, CoVe uses a neural machine translation encoder to provide contextualized representations; but in tasks that involve direct comparisons, ELMo performs better than CoVe. In (Ravishankar et al., 2021) ELMo architecture is adopted in Multilingual pre-trained language models, which include a critical step called corpus sampling to balance signal contributions from languages with adequate and inadequate resources during training. (Ravishankar et al., 2021) Compare the performance differences between more extensive multilingual language models and monolingual models for each language, and examine the effects of different corpus size ratios on subsequent performance. Because there are so many choices, choosing the best journal for study might be difficult. (Hemila and Ro"lke, 2023) investigates the use of NLP and ELMo feature engineering in content-based journal recommendation systems. Promising results were found in experiments conducted on datasets spanning physics, chemistry, and biology, with over 750,000 publications, and achieved 83% accuracy.

3.4 Universal Language Model Fine-Tuning (ULMFiT)

When it comes to interpreting language, ULMFiT (Howard and Ruder, 2018) excels at recognizing syntactic and semantic features, hierarchical relationships, and long-term dependencies. By introducing a fine-tuning transfer technique, ULMFiT avoids task-specific alterations and enables training from start without requiring large amounts of

data and time, unlike ELMo and CoVe, which fix pre-trained embeddings. The technique uses Slanted Triangular Learning Rates and discriminative fine-tuning to customize parameters based on task-specific characteristics by altering the learning rate of each layer. Gradual unfreezing, starting from the least general knowledge layer, prevents catastrophic forgetting hence these techniques collectively contribute to ULMFiT's superior performance. For effective sentiment analysis on Twitter (AlBadani et al., 2022) proposed method combined ULMFiT with Support Vector Machine (SVM). This approach enhanced the efficiency of detection and achieved the highest performance, exemplified by a remarkable accuracy of 99.78% on the Twitter US Airlines dataset.

3.5 Generative Pre Training (GPT)

With the advent of GPT (Radford, 2018), a semi-supervised method for language understanding was introduced, combining supervised fine-tuning (Eq 6) with unsupervised pre-training (Eq 5). Given an unsupervised corpus of tokens $U = \{u_1, \dots, u_n\}$, GPT uses standard language modeling objective to maximize the following likelihood:

$$L_1(U) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \Theta) \quad (5)$$

After training the model with the objective in Eq 5, the GPT adapts the parameters to the task that is supervised. By assuming labeled dataset C , where each instance comprises a sequence of input tokens, x^1, \dots, x^m along with a label y . Thus gives the following objective to maximize:

$$L_2(C) = \sum_{(x,y)} \log P(y | x^1, \dots, x^m) \quad (6)$$

Because GPT's unique structure does not require that the target task domain match the unlabeled corpus used for pre-training, it offers flexibility. The large, continuous text in GPT's pre-training dataset enables the model to comprehend long-range dependencies. Its autoregressive architecture produces a significant amount of machine-generated text, demonstrating its ability to accurately mimic human language through deep learning techniques (Kapoor et al., 2022). Based on a transformer architecture, GPT uses unsupervised learning to pre-train on extensive textual data, gaining the capacity to predict words based on contextual cues. The model's versatility is demonstrated by fine-tuning it for tasks such as text generation or classification, which entails modifying parameters for best job performance.

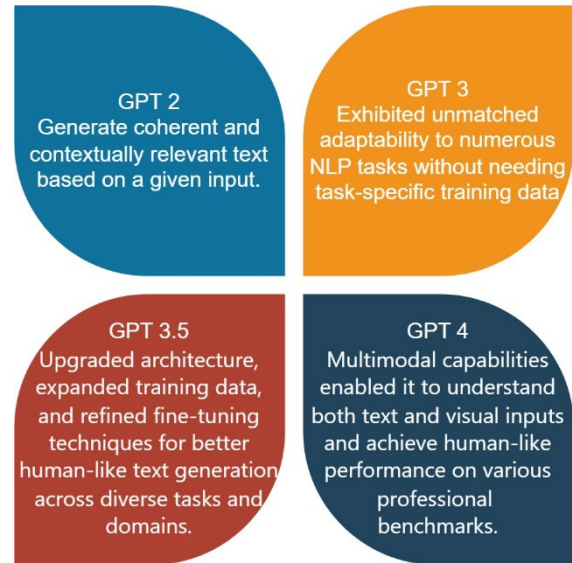


Figure 2: Versions of GPT.

Figure 2 With 117 million parameters, the first version of GPT-1 showcased the efficacy of pre-training on huge text corpora by obtaining notable results across various NLP tasks. Building on this, GPT- 2 (Radford et al., 2019) which was released with 1.5 billion parameters showed enhanced capacity to produce comprehensible text, raising initial worries about possible abuse. The next big step, GPT-3 (Brown, 2020), with 175 billion parameters, showed unmatched adaptability to various NLP tasks without requiring task-specific training data. GPT-3's cutting-edge capabilities, like few-shot and multi-task learning, have made it incredibly versatile for a range of uses, including chatbots and code generation. Language modeling activities were further optimized by the 13 Transformer components that make up the GPT-3.5 (GPT-3.5, 2023) design. GPT-4 (GPT-4, 2023), the most recent accomplishment of OpenAI, is a sizable multimodal language model capable of understanding both text and visual inputs exhibiting performance at a level comparable to that of humans on professional benchmarks.

3.6 Bidirectional Encoder Representations from Transformers (BERT)

Pre-trained word embeddings provide significant advantages over learning from scratch. BERT, as introduced in (Devlin et al., 2018), utilizes a bidirectional Transformer encoder, considering both left and right context in each layer, for predicting

masked words. In pre-training, tasks like Masked Language Model (MLM) and Next Sentence Prediction (NSP) empower BERT to learn from data that is not labeled. MLM helps integrate left and right context, addressing unidirectionality constraints. Fine-tuned with labeled data, BERT's bidirectional pre-training is crucial for applications such as question answering. Its multi-layer bidirectional transformer encoder outperforms models like GPT and ELMo, revolutionizing language understanding in NLP applications. Ongoing research focuses on refining BERT's objective and architecture.

With more significant pretraining process improvements, such as bigger batch sizes, adjusted hyperparameters, and longer training durations, are incorporated into RoBERTa, an upgraded version of BERT created by Facebook AI, leading to a language model that is more precise and resilient (Liu et al., 2019). Overcoming BERT and outperforming excelling across diverse NLP benchmarks, RoBERTa has proven to be a reliable model suitable for variety of NLP tasks, including text summarization, named entity recognition, machine translation, text classification, and question answering.

To enhance training speed and minimize memory usage, another variant of BERT known as ALBERT (Lan et al., 2019) presents two methods for parameter reduction: factorized embedding parameterization and parameter sharing across layer. It critiques the NSP goal for merging topic and coherence prediction and matching passages from various publications. By eliminating two consecutive segments and reversing the order of two consecutive segments from the same document, ALBERT chooses to use a sentence-order prediction (SOP) objective. This allows it to extract positive instances from the input.

3.7 Unified Pre-Trained Language Model (UNILM)

The constraints of BERT in natural language generation (NLG) tasks are addressed by UNILM, which was proposed in (Dong et al., 2019). UNILM is a multi-layer Transformer network optimized for unsupervised language modeling goals, in contrast to BERT, which is mostly used for natural language understanding (NLU). By using parameter sharing, its representations are more versatile, extending their applicability to a broader spectrum of uses, including

NLU and NLG. Sequence-to-sequence learning with masking is the method used by UNILM to regulate contextual attention in pretraining. Contextualized representations are learned, and subsequently fine-tuned on task-specific data for tasks downstream, providing contextual text representations that are better than their unidirectional counterparts. Based on (Dong et al., 2019) the (Zhang and Li, 2022) proposed a semisupervised way for math word problem (MWP) tasks by incorporating unsupervised pretraining and supervised tuning methods. The proposed model demonstrates improved performance over conventional models, achieving a peak accuracy of 79.57% on MWP tasks involving over 20,000+ mathematical questions.

3.8 Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA)

To improve computational efficiency and model performance, Google Research's ELECTRA (Clark, 2020) pre-trained transformer model includes a novel "replaced token detection" training method. In contrast to BERT, ELECTRA trains a discriminator to predict the replacements of some input tokens by substituting them with logical options from a generator network. The learnt representations from the generator are then improved in downstream tasks by using the pre-trained discriminator for fine-tuning. In the realm of sentimental analysis task (Zhang et al., 2022) proposed an improved approach for analyzing sentiments in Chinese short comments. The method involves integrating ELECTRA with a hybrid neural network to enhance emotional feature capture and classification accuracy.

3.9 Text-to-Text Transfer Transformer (T5)

T5, a comprehensive language model that smoothly incorporates natural language generation and processing, is presented by (Raffel et al., 2020). The adaptable T5 model finds utility in different NLP applications, serving purposes such as summarization, classification, and translation. It uses an encoder-decoder framework with a text-to-text format and creative pre-training on the Colossal Clean Crawled Corpus and during fine-tuning, T5 utilizes task-specific prefixes and adapts the token vocabulary of the decoder. With this, the T5 has proven to perform exceptionally well on NLP

benchmarks such as SQuAD, SuperGLUE, and GLUE. For detecting and quantifying gender biases in the embeddings of T5 and mT5 models. (Katsarou et al., 2022) demonstrated that employing a consistent gender direction in the mutable embedding space of Transformers, such as T5, provides a robust method for measuring biases, revealing associations between higher-status professions and a stronger association with the male gender.

3.10 XLNet

XLNet (Yang, 2019) is an autoregressive language model that uses the Permutation Language Model (PLM), a permutation-based training technique, to overcome the shortcomings of models such as BERT and GPT. For applications like sentiment analysis and text production, XLNet maintains an efficient training process by integrating auto-encoding and auto-regressive methodologies to capture bidirectional context. In order to highlight the significance of qualitative citation effect analysis, (Mercier et al., 2020) introduces ImpactCite, an XLNet-based technique that performs better than previous methods in terms of citation intent and sentiment categorization, with F1-score improvements of 3.44% and 1.33%, respectively. A novel Citation Sentiment Corpus is also provided by the authors for accurate citation sentiment classification.

3.11 Masked Sequence to Sequence (MASS)

(Song et al., 2019) MASS, a transformer-based model designed for sequence-to-sequence tasks, undergoes pre-training on the WMT monolingual corpus using a masked Sequence to Sequence approach. During this process, MASS simultaneously trains both its encoder and decoder components. Specifically, MASS utilizes masked language modeling objectives where it predicts masked tokens in the input sequence while ensuring the encoder and decoder comprehend the significance of unmasked tokens. This dual training method helps MASS effectively capture and generate meaningful representations of text in various natural language processing tasks. It attains cutting-edge performance on tasks involving the generation of language, encompassing text summarization, generation of conversational responses, and the neural machine translation. Unlike BERT, MASS is well-suited for natural language generation, demonstrating significant improvements over baselines in zero/low-resource language generation tasks.

3.12 Bidirectional and Auto-Regressive Transformers (BART)

BART, introduced by (Lewis et al., 2019), enhances pre-training for sequence-to-sequence models by employing various noising functions beyond MLM. The model corrupts input sequences using functions like sentence shuffling, deletion, token masking, text in-filling, and document rotation, achieving optimal performance with a combination of sentence shuffling and text infilling. BART demonstrates competitive results on GLUE and SQuAD, matching RoBERTa’s performance, and excels in various text generation tasks. Additionally, BART is recognized for its robust word embedding capabilities, contributing to state-of-the-art performance. (Huang et al., 2021) proposed a novel semantic sentence embedding model ParaBART, to effectively disentangles semantics and syntax by leveraging paraphrase pairs. ParaBART enhances robustness against syntactic variation in downstream semantic applications by outperforming existing models on unsupervised semantic similarity tasks. Table 1 provides a comprehensive overview of the diverse application areas where some of the embedding techniques discussed, find utility in text analysis tasks. From sentiment analysis to text mining and beyond, the table sheds light on the breadth of domains where these techniques are applied, enhancing our understanding of their versatility and impact in the domain of natural language processing.

Table 1: Embedding techniques and their applications.

Embedding Techniques	Ref (year)	Application Area	Outcomes
Context2Vec	(Babu and Kumar, 2022)	Information Retrieval	Utilizing learning algorithms, achieved a 57% accuracy in identifying threats within the darknet forums, with 89% of postings categorized as non-urgent or non-quasi
ELMo	(Altami et al., 2024)	Paraphrase	Successfully achieved a 95.68% accuracy in detecting duplicate or identical questions on the Quora platform.
	(Jain and Kashyap, 2024)	Sentiment	Developed an enhanced word vector space and a deep learning-based hybrid model, achieving a sentiment analysis accuracy of 91.64% on

ULMFiT	(Rani et al., 2024)	Sentiment Analysis	COVID-19 related Hindi tweets. Successfully identified users' sentiments toward specific products and services based on their tweets with an accuracy of 98.7%.				utilizing Transfer Learning with a tailored Unified Text-to Text Transformer T5 model, yielding F1-measures of 62.84% for ROUGE1, 54.84% for ROUGE2, and 61.98% for ROUGEL.
GPT	(Zhang Et al., 2024)	Biomedical Text Mining	Explored and demonstrated the effectiveness of fine-tuned large language models (LLMs) for complex chemical text mining tasks, achieving accuracies ranging from 69% to 95%.			Text Emotion Recognition	Leveraged XLNet for contextual information learning, bidirectional gated recurrent unit (BiGRU) for feature extraction, and attention mechanism for enhancing important information, resulting in improved word vectors quality and sentiment analysis model judgments accuracy with an achieved accuracy of 91.71%.
	(Bang et al., 2023)	Sentiment analysis, misinformation	Assessed ChatGPT's performance across multiple tasks, languages, and modalities, achieving an average accuracy of 63.41% across 10 different reasoning categories encompassing logical, non-textual, and common sense reasoning.				
BERT	(Kumar and Solanki, 2023)	Named Entity Recognition	Improved natural language understanding by training a BERT model to accurately identify and classify named entities (persons, places, organizations, time, money, etc.) within the text, achieving a remarkable accuracy of 98.52% on the English CoNLL-2003 NER dataset.	BART	(Raju et al., 2024)	Text Error Analysis and Correction	Analyzed various errors in text documents and utilized advanced deep neural network-based language models, BART and MarianMT, to rectify these anomalies. BART exhibited superior performance in handling spelling errors, achieving a reduction of 24.6%, while in grammatical errors, it achieved a reduction of 8.8%.
ELECTRA	(C, epni et al., 2023)	Text Classification	Successfully classified texts extracted from the website URLs into their respective industries with an accuracy of 98%.				
	(Mala et al., 2023)	Sentiment Analysis	Conducted sentiment analysis on movie reviews utilizing various transformer models, achieving an accuracy of 93.32%.				
T5	(Ismail et al., 2023)	Text Summarization	Investigated automatic text summarization				

4 CONCLUSIONS

This review paper explores into a broad spectrum of pre-trained language models, highlighting their advancements and applications in NLP. From conventional models like Word2Vec and GloVe to state-of-the-art transformers such as GPT-3 and BERT, each model is analyzed for its strengths and limitations. The paper emphasizes the evolution from non-contextualized to contextualized embeddings, showcasing the significance of capturing both syntactic and semantic subtleties. The study introduces innovative models like ParaBART, which effectively disentangles semantics and syntax,

outperforming existing models on unsupervised semantic similarity tasks. The review covers various techniques, including contextualized embeddings like ELMo, CoVe, and UNILM, each contributing to improved performance on downstream NLP tasks. Moreover, the paper discusses models designed for specific purposes, such as ELECTRA for enhanced computational efficiency and T5 for versatile text-to-text transfer capabilities. The importance of fine-tuning techniques, like ULMFiT, is highlighted for achieving superior task-specific performance. Hence, the diverse landscape of pre-trained language models offers a wealth of opportunities for advancing NLP tasks. The continuous evolution from non-contextualized to contextualized embeddings, along with innovations in model architectures and training methods, contributes to the growing success and applicability of these models in understanding and generating human-like language. In the course of research, it was discovered that relying on a single method for evaluating the semantic and sentiment analysis of text may not be sufficient. The evolution of word embeddings has aimed to provide more straightforward explanations of textual data, addressing the increasing challenges in NLP. Future studies might explore advanced methods or develop sophisticated models to further enhance our understanding and capabilities in handling complex linguistic tasks.

ACKNOWLEDGEMENTS

The author would like to express sincere gratitude to Mr. Puneet Kapoor, Dr. Sakshi Kaushal and Ms. Ishani Sharma for their support and valuable suggestions that helped to improve and evaluate this paper.

REFERENCES

- AlBadani, B., Shi, R., and Dong, J. (2022). A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM. *Applied System Innovation*, 5(1):13.
- Altamimi, A., Umer, M., Hanif, D., Alsubai, S., Kim, T.-H., and Ashraf, I. (2024). Employing Siamese MaLSTM model and ELMo word embedding for Quora duplicate questions detection. *IEEE Access*.
- Ashihara, K., Kajiwar, T., Arase, Y., and Uchida, S. (2018). Contextualized word representations for multi-sense embedding. In *Proceedings of Pacific Asia Conference on Language, Information and Computation*, volume 32, pages 1–9. Association for Computational Linguistics.
- Ashihara, K., Kajiwar, T., Arase, Y., and Uchida, S. (2019). Contextualized context2vec. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 397–406. Association for Computational Linguistics.
- Asudani, D. S., Nagwani, N. K., and Singh, P. (2023). Impact of word embedding models on text analytics in deep learning environment: a review. *Artificial intelligence review*, 56(9):10345–10425.
- Babu, J. S. and Kumar, A. P. (2022). Comprehensive self dark web segmentation for cyber security risk intelligence. *International Journal of Conceptions on Computing and Information Technology*, 8.
- Bang, Y., Cahyawijaya, S., Lee, N., Dai, W., Su, D., Wilie, B., Lovenia, H., et al. (2023). A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.
- Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.
- Brown, T. B. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
- Clark, K. (2020). Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., and Hon, H.-W. (2019). Unified language model pre-training for natural language understanding and generation. *Advances in Neural Information Processing Systems*, 32.
- GPT-3.5(2023).Gpt-3.5 documentation. https://huggingface.co/transformers/v3.5.1/model_doc/gpt.html. Available online, Accessed on December 2023.
- GPT-4 (2023). Gpt-4 paper. <https://cdn.openai.com/papers/gpt-4.pdf>. Available online, Accessed on December 2023.
- Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.
- Han, T., Zhang, Z., Ren, M., Dong, C., Jiang, X., and Zhuang, Q.-s. (2023). Text emotion recognition based on xlnet-bigru-att. *Electronics*, 12(12).
- Harris, Z. S. (1954). *Distributional Structure*. Word.
- Hashempour, R. and Villavicencio, A. (2020). Leveraging contextual embeddings and idiom principle for detecting idiomaticity in potentially idiomatic expressions. In *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 72–80. Association for Computational Linguistics.

- Hemila, M. and Ro"lke, H. (2023). Recommendation system for journals based on ELMo and deep learning. In 2023 10th IEEE Swiss Conference on Data Science (SDS), pages 97–103. IEEE.
- Hofmann, V., Pierrehumbert, J. B., and Schu"tze, H. (2020). Dynamic contextualized word embeddings. arXiv preprint arXiv:2010.12684.
- Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146.
- Huang, J. Y., Huang, K.-H., and Chang, K.-W. (2021). Disentangling semantics and syntax in sentence embeddings with pre-trained language models. arXiv preprint arXiv:2104.05115.
- Ismail, Q., Alissa, K., and Duwairi, R. M. (2023). Arabic news summarization based on t5 transformer approach. In 2023 14th International Conference on Information and Communication Systems (ICICS), pages 1–7. IEEE.
- Jain, V. and Kashyap, K. L. (2024). Enhanced word vector space with ensemble deep learning model for covid-19 hindi text sentiment analysis. Multimedia Tools and Applications, pages 1–22.
- Kapoor, P., Kaushal, S., and Kumar, H. (2022). A review on architecture and communication protocols for electric vehicle charging system. In Proceedings of the 4th International Conference on Information Management Machine Intelligence, pages 1–6.
- Katsarou, S., Rodr"iguez-Ga"lvez, B., and Shanahan, J. (2022). Measuring gender bias in contextualized embeddings. Computer Sciences and Mathematics Forum, 3(1):3.
- Katz, S. (1987). Estimation of probabilities from sparse data for the language model component of a speech recognizer. IEEE Transactions on Acoustics, Speech, and Signal Processing, 35(3):400–401.
- Kim, T., Choi, J., and Lee, S.-g. (2018). Snu ids at semeval 2018 task 12 sentence encoder with contextualized vectors for argument reasoning comprehension. arXiv preprint arXiv:1805.07049.
- Kumar, S. and Solanki, A. (2023). Named entity recognition for natural language understanding using Bert model. In AIP Conference Proceedings, volume 2938. AIP Publishing.
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite Bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.
- Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized Bert pre-training approach. arXiv preprint arXiv:1907.11692.
- Mala, J. B., Angel SJ, A., Raj SM, A., and Rajan, R. (2023). Efficacy of electra-based language model in sentiment analysis. In 2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICIS-CoIS), pages 682–687. IEEE.
- Mars, M. (2022). From word embeddings to pre-trained language models: A state-of-the-art walkthrough. Applied Sciences, 12(17):8805.
- McCann, B., Bradbury, J., Xiong, C., and Socher, R. (2017). Learned in translation: Contextualized word vectors. Advances in Neural Information Processing Systems, 30.
- Melamud, O., Goldberger, J., and Dagan, I. (2016). context2vec: Learning generic context embedding with bidirectional LSTM. In Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, pages 51–61. Association for Computational Linguistics.
- Mercier, D., Rizvi, S. T. R., Rajashekar, V., Dengel, A., and Ahmed, S. (2020). Impactcite: An Xlnet-based method for citation impact analysis. arXiv preprint arXiv:2005.06611.
- Mikolov, T. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 3781.
- Neelima, A. and Mehrotra, S. (2023). A comprehensive review on word embedding techniques. 2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS), pages 538–543.
- Patil, R., Boit, S., Gudivada, V., and Nandigam, J. (2023). A survey of text representation and embedding techniques in nlp. IEEE Access, 11:36120–36146.
- Pennington, J., Socher, R., and 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. Association for Computational Linguistics.
- Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, volume 1, pages 2227–2237. Association for Computational Linguistics.
- Radford, A. (2018). Improving language understanding by generative pre-training. arXiv preprint.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8):9.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140):1–67.
- Raju, R., Pati, P. B., Gandheesh, S. A., Sannala, G. S., and Suriya, K. S. (2024). Grammatical versus spelling error correction: An investigation into the responsiveness of transformer-based language models using Bart and MarianMT. Journal of Information & Knowledge Management.

- Rani, L. S., Zahoor-Ul-Huq, S., and Shoba Bindu, C. (2024). A deep learning approach for Twitter sentiment analysis using ULM-SVM. In 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom), pages 1639–1643. IEEE.
- Ravishankar, V., Kutuzov, A., L., and Velldal, E. (2021). Multilingual ELMo and the effects of corpus sampling. In Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa), pages 378–384. Association for Computational Linguistics.
- Salton, G., Wong, A., and Yang, C.-S. (1975). A vector space model for automatic indexing. Communications of the ACM, 18(11):613–620.
- Shi, Y., Yao, K., Tian, L., and Jiang, D. (2016). Deep LSTM-based feature mapping for query classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1501–1511. Association for Computational Linguistics.
- Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. (2019). Mass: Masked sequence to sequence pre-training for language generation. arXiv preprint arXiv:1905.02450.
- Wang, T., Chen, P., Amaral, K., and Qiang, J. (2016). An experimental study of LSTM encoder-decoder model for text simplification. arXiv preprint arXiv:1609.03663.
- Yang, Z. (2019). Xlnet: Generalized autoregressive pre-training for language understanding. arXiv preprint arXiv:1906.08237.
- Zhang, D. and Li, W. (2022). An improved math word problem (MWP) model using unified pre-trained language model (UniLM) for pretraining. Computational Intelligence and Neuroscience, 2022(1):7468286.
- Zhang, S., Yu, H., and Zhu, G. (2022). An emotional classification method of Chinese short comment text based on ELECTRA. Connection Science, 34(1):254–273.
- Zhang, W., Wang, Q., Kong, X., Xiong, J., Ni, S., Cao, D., Niu, B., et al. (2024). Fine-tuning large language models for chemical text mining. ChemRxiv.
- Cepni, S., Toprak, A. G., Yatkinoglu, A., Mercan, B., and Ozan, (2023). Performance evaluation of a pre-trained Bert model for automatic text classification. Journal of Artificial Intelligence and Data Science, 3(1):27–35.