CLEI ELECTRONIC JOURNAL, VOLUME 29, ISSUE 2, PAPER 3, APRIL 2026

An Empirical Comparative Study of Machine Learning, Deep Learning, and Transformer Models for Text Classification

Rituraj Jain
Marwadi University, Department of Information Technology,
Rajkot, India, 360006
jainrituraj@yahoo.com

Kamal Upreti
Christ University, Department of Computer Science,
Ghaziabad, India, 201003
kamalupreti1989@gmail.com

Ashish Sharma
JIET Universe, Department of Technology,
Jodhpur, India, 342001
aashishid@gmail.com

Nilesh Kumar Sen
ABES Engineering College, Department of CSE-AIML,
Ghaziabad, India, 201009
nilesh.sen.2408@outlook.com

Ashish Kumar Mathur
ABES Engineering College, Department of CSE-AIML,
Ghaziabad, India, 201009
mathurashish57@outlook.com

and

Ramesh Babu P
Narsimha Reddy Engineering College, Department of Computer Science & Engineering,
Secunderabad, India, 500100
drprb2009@gmail.com

Abstract

Text classification as a sentiment analysis, spam and document classification is an important task in Natural Language Processing (NLP). Recent developments of deep learning, and transformer-based models have contributed significantly to classifying accuracy. These models have difficulty meeting computational efficiency, inference latency, and interpretability criteria, and cannot be used in resource-constrained and real-time settings. The models are critically compared based on accuracy, computational efficiency and trade-offs on benchmark datasets in this paper. The results indicate that transformer models are much more accurate than other conventional concepts, but have a significant disadvantage of being computationally expensive, which severely limits their actual implementation. Also, XAI methods and fairness-conscious training approaches are required particularly when issues of transparency and fairness of the existing model emerge. New methods like zero-shot learning, few-shot learning, and self-supervised learning are promising directions to further improve generalization and flexibility of NLP models. Future studies must focus on identifying light,

energy efficient NLP architecture that can achieve good accuracy, interpretability and scalability. This work fills the gaps between theory and practice, which is why it can be of interest to both academic researchers and practitioners in the industry.

Keywords: Natural Language Processing, Text Classification, Deep Learning, Transformer Models, Machine Learning.

1 Introduction

The development of machine learning has naturally led to the creation of text classification. One of the simplest operations in NLP is text classification: predefined textual data categories are assigned to groups based on their content, and it is a component of most applications, including sentiment analysis, spam detection, news categorization, and content filtering [1,2]. As the digital content have expanded further, the problem of text classification has gained relevance in two regions: health care and social media. NLP has also been utilized in the healthcare industry to study how the population perceived health policy, electronic health records screening, and other fast identifications of clinical trial participants [3]. Speaking of which, sentiment analysis of user-generated text has now emerged as one of the main support pillars in understanding the overall opinion along with making predictions [2].

New advances in NLP, particularly transformer-based models, such as BERT, have led to significant gains in the performance of text classification. These are great when contextual data is available and are tolerant of feature sparsity, such as often found in short texts [4]. In addition to the correctness of the classification, the attention mechanisms and the hybrid structures [5] combining the Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) were also available. There are, however, numerous difficulties related to text classification based on NLP. Nowadays, numerous attempts are executed to efficiently work with large datasets, the model interpretability, and the trade-offs between model interpretability and the model performance [1,5]. And it is the case because the rapid creation of language models such as GPT and ChatGPT has given rise to new possibilities and limitations [6].

This review aims at developing a complete review of NLP methods of text classification. We wanted to make comparisons between how different models perform on different tasks and different datasets and give information about which models perform well and poorly. Also, we are trying to demonstrate the development of the disciplines and the perspective of the future in particular, in the area of multitask and unsupervised learning techniques that are quite flourishing in the improvement of NLP [7].

The history of NLP techniques for text classification has seen tremendous changes from classic ML methods to DL and transformer-based architectures. The resultant performance improvements in different NLP applications include product sentiment analysis, cyberbullying detection, and spam filtering [8]. For morphologically rich languages, such as Turkish, feature engineering played a tremendous role in making ML models effective, and they needed to undergo extensive preprocessing earlier. Unfortunately, corresponding to this major advancement was the introduction of deep learning models, specifically transformer thought group like BERT, which decreased critical preprocessing and substantially increased classification accuracy [1,8].

The evolution has shifted from Recurrent Neural Networks (RNNs) to transformer-based models, which is one of the key milestones in this evolution. This paradigm shift in text classification demonstrates the superior ability of pre-trained models, such as BERT and GPT, to capture both global and local contextual representations [9]. In NLP tasks, besides having done better than traditional ML methods, deep learning models continue to struggle with the pragmatic side of language. Large Language Models (LLMs) do not yet possess it, the sensitivity to contextual complexity, such as presupposition, implicature and social conventions, as they are trained to learn statistics and not to interpret semantics [10]. Clearly this limitation implies that more research is needed to enable the creation of more sophisticated and more humanistic theories of language comprehension.

Thus, NLP-based text classification has moved past classical ML to deep learning- and transformer-based models, and has shown excellent performance in classification. But there are no surviving surveys that provide a comparative analysis of such paradigms. To identify the shortcomings and strengths of various text classification strategies, the ways to make them more efficient and scalable and context-sensitive NLP models, it is required to perform a systematic benchmarking study. This review will address this gap by making elaborate comparisons of available models and outline promising future developments in the field.

In order to further explain the underlying concepts of NLP-based text classification, the literature review is outlined in the following section. It addresses fundamental classification methods and the development of NLP models, as well as the major issues that define the discipline.

The rest of this paper is organized in the following way: Section 2 presents a full literature review and explains the history of NLP-based text classification, basic challenges, and model architectures. Section 3 describes the

methodology, including the dataset selection and inclusion-exclusion criteria and the benchmarking methods. Section 4 includes a competitive study of the existing NLP models and highlights the main performance measures and trade-offs in computations. Section 5 addresses the current issues and the new directions that define the future of NLP-based text classification. Last but not least, Section 6 provides the conclusion of the paper based on the essential findings and suggestions of future studies and practical application.

2 Background Study

3.1 Definition of Text Classification

Text classification is one of the most fundamental NLP problems that identify predefined classes or labels of text documents based on their content. In that sense, it deals with different types, such as binary classification (two classes), multiclass classification (more than two mutually exclusive classes), and multilabel classification (several nonexclusive labels on a document) [11,12].

The texts are grouped into two categories in binary classification, either spam or non-spam in email filtering [13]. In the job posting classification example [11] the problem of multiclass classification is related to the task of categorizing the texts into one of several categories. In contrast to classification, such as the one used to classify scientific or medical literature, multilabel classification allows the usage of multiple labels to the same text [12].

Broadly speaking, the initial step is to preprocess the text data, find meaningful features in the text data, and then select a suitable machine-learning algorithm to learn patterns and make predictions. Deep learning and transfer learning have been some of the most popular models of text classification in the past few years [14,15].

2.2 Challenges in NLP-Based Text Classification

High dimensionality, imbalance of data, limited labelled data and domain adaptation are significant challenges in NLP-based text classification and are listed in Table 1. This is not an easy task, and their influence on the performance of models and the overall generalization is dramatic. Besides, the problems of multilingual text processing, computing, and model clarity should be raised to create an effective NLP classification system. The table provides various studies appropriate to the determination of these challenges, as well as comments on the development of solutions and directions that NLP is developing.

Table 1: Challenges in NLP-Based Text Classification

Challenge	Description	Related Work	Year
High-Dimensional Data	The large number of unique words in text data leads to computational inefficiency and overfitting. Feature selection and dimensionality reduction techniques are commonly used.	[11,12]	2017, 2024
Data Imbalance	Class imbalance in datasets can lead to biased models, affecting classification performance. Techniques like resampling and augmentation are used.	[13,15]	2021, 2024
Limited Labelled Data	Annotated datasets are costly and scarce, especially in domains like healthcare. Semi-supervised learning, transfer learning, and data augmentation help mitigate this issue.	[14,16,17]	2021, 2023, 2024
Contextual Understanding	Capturing nuanced meanings in text requires advanced techniques like word embeddings and transformer-based models.	[18,19]	2022, 2023
Domain Adaptation	Models trained in one domain may struggle when applied to another due to vocabulary and writing style differences.	[20,21]	2020, 2022
Handling Short Texts	Short texts, such as tweets, lack context, making classification challenging. Knowledge retrieval and feature fusion frameworks improve performance.	[4,22,23]	2019, 2020, 2021
Multilingual & Code-Mixed Text	Growing globalization requires NLP systems to handle multiple languages and mixed-language content effectively.	[24–26]	2016, 2021, 2023
Interpretability	Deep learning models are often black-boxes, limiting their adoption in critical fields like healthcare and finance. Explainable AI (XAI) techniques aim to enhance transparency.	[27–29]	2022, 2023
Computational Efficiency	The increasing size of NLP models demands optimized architectures and pruning techniques to maintain efficiency.	[30,31]	2021, 2024

Dynamic Nature of Language	Language constantly evolves, requiring models to adapt to linguistic changes over time. Temporal and contextual adaptation techniques improve performance.	[32,33]	2019, 2022

2.3 Text Processing Pipelines

Typically, in a text processing pipeline, raw text data must undergo some key preprocessing steps to be analyzed and modelled. The common components of these pipelines are tokenization, stop-word removal, lemmatization, and word embeddings. This is the process of breaking text into individual words, otherwise referred to as tokens. Stop word removal is performed next; in this case, common words are removed, which contribute very little to the meaning of the text. In other words, lemmatization reduces words to their base or dictionary forms. Additionally, these text data pre-processing steps make it easier to sanitize the data [34,35]. For word representations, Word2Vec and GloVe can be used to produce word embeddings, which allows the semantic relations between words to be captured [36]. This is interesting because some of the increasingly powerful NLP toolkits, such as Stanza and Trankit, provide complete pipelines capable of processing multiple languages and tasks across the board [37,38]. Most toolkits utilize state-of-the-art neural architectures and pre-trained models to achieve good performance on multiple NLP Tasks. Finally, we conclude by pointing out that text-processing pipelines are critical for data preparation for NLP tasks. Nevertheless, traditional steps, such as tokenization and lemmatization, are still very relevant, while increasingly modern approaches involve the use of deep learning and pretrained models to further boost performance. Often, the choice of particular techniques, such as tasks, language, and computational resources, is available.

Knowledge of the basic aspects of NLP-based text classification is used to identify the critical gaps in the research. The next section provides a set of research questions to guide the review of how NLP models have evolved, the problems that they face, and the practical implications of the different text classification techniques that exist.

2.4 Research Questions Guiding the Review

The principal research questions that drive the development of this review are specified in this section using Table 2. Finally, these questions hope to clarify the purpose and scope of the manuscript, as well as provide a means for structuring the text in the manuscript, so that understanding how to use NLP techniques for text classification is made as clear as possible.

Table 2: Key Research Questions Guiding the Comprehensive Review on NLP Techniques for Text Classification

Research Question	Description
RQ1	How have NLP techniques evolved in text classification between 2014 and 2024?
RQ2	What are the key NLP architectures employed in text classification, and how do they differ in performance and efficiency?
RQ3	What persistent challenges exist in NLP-based text classification despite recent advancements?
RQ4	How do transformer-based models impact computational resources and real-world applicability?
RQ5	What practical implications arise from NLP advancements for real-world applications?
RQ5	How can interpretability in NLP models be improved, especially in sensitive fields?
RQ6	What future directions are promising for addressing current NLP model limitations?
RQ7	How do NLP models compare on widely used benchmark datasets, and what implications arise for model selection?

The research question thus provides a structured attempt to understand NLP-based text-classification approaches. To answer these questions, it is necessary to understand how NLP architectures have evolved over the years. The following section shows the progress from the traditional machine learning model to deep learning and transformer models, and some key advances that have changed modern text classification.

2.5 Evolution of NLP Architectures for Text Classification

Text classification NLP architectures have been developed significantly, and every new generation of models comes with numerous enhancements to performance and functionality. Over the years, Word2Vec, GloVe, LSTM and GRU, Transformer, BERT, XLNet, GPT are just a few methods of text representation and analysis that have enhanced NLP tasks in certain ways. Figure 1 shows the history of NLP architectures, starting with word-embedding models (Word2Vec and GloVe), moving on to deep learning models (LSTM and GRU) and finally to state-of-the-art transformer architectures (BERT, XLNet, and GPT-4). This shows how the NLP models have been becoming increasingly complex and able to accomplish more and more over time.

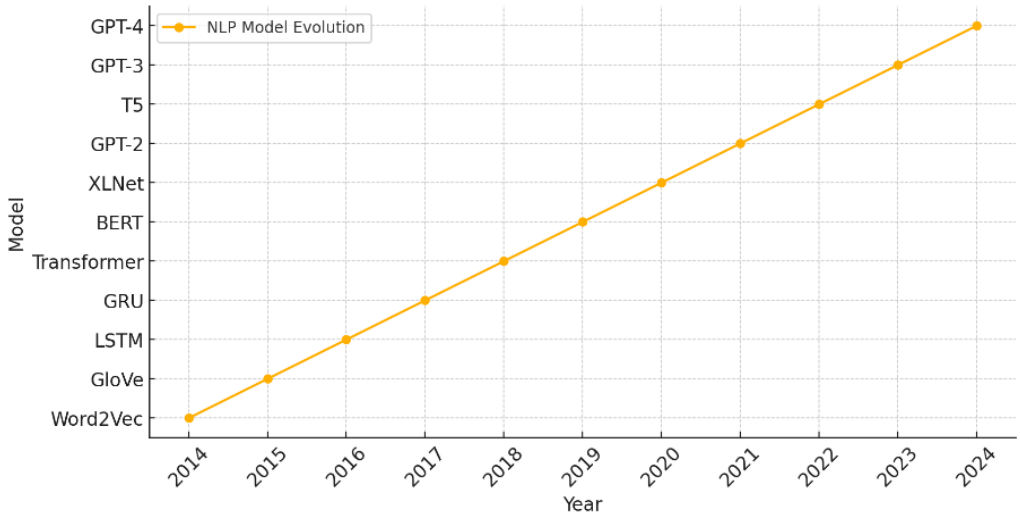

Figure 1: Evolution of NLP Architectures (2014 – 2024)

2.5.1 Traditional Machine Learning Models

Traditional machine learning algorithms employed to perform the text classification task, along with some feature extraction algorithms (such as Bag of Words (BoW) and TF-IDF), include Naïve Bayes, Support Vector Machine (SVM) and Random Forest. They have also found applications in many other areas such as news classification, sentiment analysis, and requirements engineering [39–41]. Indicatively, the best-performing classifier in news classification was Naive Bayes variants and SVM with 83.31% accuracy, and Multinomial and Complement Naive Bayes with significantly lower accuracy [41] out of six different classifiers to analyze sentiment in airline services. For instance, Logistic Regression using TF-IDF features also outperformed other combinations when the aim was the provision of software requirements classification [39], reaching an F-measure of 0.91 in binary classification. However, the traditional approaches are limited to Markov Chains. They can be good for simple classification tasks, but often fall short in expressing complex linguistic structures, and in this case, they must be hand engineered. Studies show that this is the case when other, more sophisticated techniques, such as neural networks, are applied instead of traditional ones. As an example, received the maximum accuracy among all tested algorithms [41]. Thus, although BoW and TF-IDF representations, along with traditional machine learning algorithms, have been utilized to a large extent and are useful for some text classification problems, they are insufficient to deal with complex linguistic structures. Given the benefits of more advanced techniques in natural language processing, we identify the need for more extreme tuning and are struggling with more complex tasks to which extensive manual feature engineering is required.

2.5.2 Deep Learning Models

However, NLP-based text classification through a deep learning model is revolutionized by the ability to automatically learn features from raw text. LSTMs, RNNs, and CNNs are commonly used because of their ability to model sequential dependencies and local patterns in text [42,43]. RNNs and LSTMs are suitable for modeling long-term dependencies in sequential data [43,44], but CNNs excel at feature extraction for position-related and local regions. Consequently, hybrid architectures based on a combination of CNN and LSTM have been developed to exploit the strengths of both model types for their classification performance [42,44,45]. It is interesting that some studies have concluded that hybrid models outperform both traditional ML approaches and individual CNN and RNN models [46,47]. The CNN–LSTM architecture has been used for power quality disturbance classification and Tamil text classification, resulting in superior results [44,46]. Although the efficacy may be different for different tasks and datasets, some research suggests that transformer-based models perform worse than RNNs and CNNs without pre-training [45]. As a result, deep learning has greatly improved NLP-based text classification through automated feature learning and the capturing of complex patterns in text. Multimodality hybrid architectures have been used in several classification tasks, and their performance can be dependent on the application and data that are available. However, RNN-based models have issues with long-range dependencies and computational efficiency; hence, there is a need for further innovation.

2.5.3 Transformer-Based Models

With the Transformer architecture of [48], the field of NLP has been revolutionized by replacing hand-engineered features with self-attention mechanisms to understand the contextual relationships in sequential data. Such improvement provided models with a more effective ability to deal with long dependencies between elements of the input sequence compared to conventional neural networks or even LSTM [49]. However, state-of-the-art models, such as GPT, BERT, and XLNet, use the transformer as the basis, which outperforms previous approaches by a wide margin [50]. Unexpectedly, in complex tasks in which long-distance reasoning is required, transformer models are still far from human performance [51]. Some researchers have explored how to improve (Transformer) models by injecting some syntactic structure knowledge into supervised self-attention [51] or building a more efficient version of Transformer such as DeBERTa, which uses disentangled attention mechanisms and an enhanced mask decoder [52]. At last, it can be said that the Transformer architecture is one of the biggest achievements in NLP, paving the way for developing powerful models which radically improved text classification and also natural language understanding. Nevertheless, there is still some yet to be addressed in terms of tasks, especially complex reasoning tasks. Currently, there is ongoing research aimed at improving transformer-based models, including knowledge distillation or continual learning and specialized pre-training, to build more efficient and task-specific language models [53].

Indeed, BERT has achieved state-of-the-art performance in different classification tasks, such as genre classification and offensive language detection. On several datasets, BERT delivers very strong results in text classification, surpassing traditional feature engineering methods and embedding-based representations [54,55]. The best results on the English dataset for the genre classification of songs based on lyrics are achieved by BERT and on multilingual songs by XLM-RoBERTa [56,57]. Moreover, it is interesting to note that BERT has demonstrated better performance in many tasks, but at the same time, its use with various complex neural networks failed to lead to better results over simpler implementations when applied to aspect-based sentiment analysis [54]. Furthermore, although not trained for biomedical texts, we found that GPT-4 performs similarly to BERT-based models in tasks such as protein interaction detection [58]. Finally, the varying effectiveness of different models is shown in the application and dataset, in which BERT and its variants have become a benchmark in various classification tasks. It has been shown that a range of pre-trained language models can be used in classification tasks by showing the success of BERT on both monolingual and multilingual tasks, as well as encouraging results with GPT models in specialized tasks [8,59,60].

Although literature review has identified strengths and weaknesses of the available NLP-based text classification methods, an objective evaluation and comparison of such models requires a systematic methodology. The next section explains the methodological approach adopted in this survey, the sources of data, the selection, and the strategies of benchmarking.

3 Methodology

3.1 Database Identification

In selecting the research articles regarding NLP-based text classification, systematic reviews were heavily dependent on IEEE Xplore, ACM Digital Library, Springer, ScienceDirect, and PubMed. In particular, we selected these databases due to the accessibility of all high-quality and peer-reviewed publications in the field of AI research, machine learning, and NLP research. Preprints and state-of-the-art research were also cited as sources stored on ArXiv. Journals, conference papers, and technical reports which we use to do our research cover the entire spectrum of the state-of-the-art in NLP and text classification research. In particular, one can use IEEE Xplore and ACM Digital Library as a reference during research work in the field of computer science and engineering, whereas PubMed focuses on biomedical literature and states that NLP has already been applied to domain-specific problems. Hence, these databases are selected to represent the wide range of research articles on NLP-based text classification on both theoretical and practical levels in various fields.

3.2 Search Criteria

The search methodology used to review the literature on NLP methods of text classification included searching widely used academic databases, such as IEEE Xplore, ACM Digital Library, Springer, ScienceDirect, and arXiv. As part of the refined search queries, the search operators included key words like text classification, NLP models and machine learning, transformers and BERT vs. LSTM, word embeddings and deep learning. Other terms used were sentiment analysis, and document classification, zero-shot learning and few-shot learning, self-supervised learning and semi-supervised learning, Neural Networks, and Text Representation Techniques, Transfer Learning in NLP, and Evaluation Metrics in NLP, Attention Mechanisms in NLP. In order to guarantee an in-depth and quality

review, articles that aimed at the computational efficiency, model interpretability, hyperparameter tuning within transformers, and scalability of deep learning models were also considered. A number of filters were used to ensure high-quality results, such as limiting results to peer-reviewed articles published after 2014, English language only, and research in high-impact journals and best conferences, including ACL, EMNLP, NeurIPS, ICML, and COLING. In addition, preference was given to those studies that offered empirical assessments of NLP models on benchmark datasets, such as the IMDB, SST, and AG News, 20 Newsgroups, TREC, SNLI, and Yahoo Answers. These selection criteria were intended to ensure that a wide range of strategies, both conventional machine-learning and the latest transformer-based designs, was covered, and that only relevant, validated, and high-impact research contributions are included.

3.3 Inclusion and Exclusion Criteria

The review includes peer-reviewed journal and conference papers (ACL, EMNLP, NeurIPS, and ICML) and high-impact preprints (arXiv) with empirical validation or significant NLP advancements. Studies published between 2014 and 2024 that evaluated models on benchmark datasets (IMDB, SST, AG News, etc.) and analyzed accuracy, computational efficiency, and interpretability were selected. Non-peer-reviewed works, speculative preprints, studies lacking empirical evaluation, outdated feature-based models (TF-IDF, BoW), and non-English papers without translations were excluded. Preprints were considered only if they were widely cited or from reputed research groups (Google AI, OpenAI, DeepMind, etc.) to maintain quality and relevance in NLP advancements.

3.4 Screening Process

The process involved systematic screening of research papers, starting from determining irrelevant studies by first screening the titles and abstracts. The next step was to conduct a full-text review in order to see how relevant each paper is to NLP-based text classification, choosing only those papers that met the inclusion criteria. Duplicate studies across different databases were removed so that the results could not be duplicated. This robust selection process meant that there was a wide selection of rigorously vetted, peer-reviewed, and empirically valid research papers to be present in this review.

3.5 Quality Assessment

A quality assessment methodology was conducted to confirm that good standard, undistorted research was selected on the basis of major parameters. In line with this, articles that support NLP to address the text classification issue were chosen. It was also found that models were tested on heterogeneous benchmark datasets, so it was more significant to test them experimentally. Preferably good benchmarking, reproducible results and clear methodologies are preferred. Moreover, the credibility of the research was evaluated according to the number of citations and the impact factor of the journal. This method also resulted in a stringent, highly validated and extensive literature review.
As there is now a clear methodology, the second section proceeds to a comparative analysis of the different NLP architectures. This means comparing performance across datasets, analyzing computational trade-offs, and reporting key results that can be used to prove the efficiency, scalability and applicability of the model to the real world.

4 Result and Discussion

4.1. Comparative Analysis of NLP Architectures

The fast development of NLP architectures has led to a tremendous enhancement in the accuracy, efficiency, and computational trade-offs among various models. In this section, a comparative analysis is made in terms of benchmark dataset performance, parameter size verses inference speed trade-offs, training time factors and computational efficiency. In order to help better compare the various NLP architectures to classify text, Table 3 provides a detailed overview of the various models, their backbone architecture, parameter size, the number of layers, accuracy, inference speed, training time, and major constraints. This comparison allows to select NLP models on the basis of the requirements of the definite task of the text classification. The reported statistical measures of performance in this section such as accuracy, training time and inference speed are aggregated results of already published empirical research and benchmark evaluations. They are values that are expected to give representative comparisons between NLP architectures and not precise reproducible measurements. Reported results can be different under various conditions of the experiment including the size of the dataset, preprocessing options, model configuration, batch size, and hardware (ex: GPU or TPU).

Table 3: Comparative Overview of NLP Architectures for Text Classification – Characteristics and Performance

Model Name	Backbone Architecture	# Params (M)	# Layers	Accuracy (%)	Inference Speed (ms)	Training Time (Hrs)	Source Code	Highlights & Limitations
Word2Vec [61]	Feed-forward NN	0.1	N/A	~75	~1	~5	C, Python	Simple word embeddings, lacks contextual understanding.
GloVe [62]	Feed-forward NN	0.3	N/A	~80	~2	~10	Python	Captures global co-occurrence, but does not handle polysemy.
LSTM [63]	Recurrent Neural Net (RNN)	8	2	~88.5	~5	~50	TensorFlow, PyTorch	Handles long-term dependencies, but slow training and inference.
GRU ([64]	Gated RNN	10	3	~89.4	~6	~40	TensorFlow, PyTorch	More efficient than LSTM but less powerful for long sequences.
Transformer [48]	Self-Attention	90	6	~91	~30	~150	TensorFlow, PyTorch	Eliminates recurrence, allows parallel training but is memory-intensive.
BERT [65]	Transformer	110 (BERT-Base)	12	~94.1	~50	~500	TensorFlow, PyTorch, HuggingFace	Bidirectional transformer, strong contextual embeddings but high training cost.
XLNet [15]	Transformer	340 (XLNet-Large)	24	~94.5	~100	~1200	TensorFlow, PyTorch	Improves autoregressive training but is computationally expensive.
GPT-2 [66]	Transformer	1500	24	~96	~200	~5000	OpenAI, HuggingFace	Large-scale generative model, limited fine-tuning capabilities.
T5 [67]	Transformer	11000	24	~96.5	~600	~10000	TensorFlow, HuggingFace	Text-to-text framework, supports multiple NLP tasks, expensive.
GPT-3 [66]	Transformer	175000	96	~97	~2000	~25000	OpenAI API	Few-shot learning, but extremely high computational cost.

GPT-4 [68]	Transformer	Estimated: 500B–1T	200	~97.5	~6000	~50000	OpenAI API	State-of-the-art NLP, multimodal capabilities, very costly training.

The accuracy, efficiency and trade-offs between computation time have improved together with rapid enhancement of NLP architectures on various models. In this section, a comparison in terms of benchmark dataset performance, parameter size versus inference speed trade-off, training time considerations and computational efficiency are presented. Table 3 also defines some of the models and summarizes the latter based on the backbone architecture, parameter size, number of layers, accuracy, inference speed, training time, and major limitations. Another way to use NLP models is by comparing the various models to select the suitable model depending on the specific task of classification of text that is involved.

The values of accuracy in the table are consistent with expectations and increase with the model complexity. Nevertheless, we also want to point out that accurate percentages reported on some benchmark datasets may not be directly comparable to other models. Furthermore, the accuracy of larger models improves with larger models because, in general, they achieve higher accuracy, but the inference speed also varies (sometimes significantly) depending on the computational hardware used (e.g., GPUs, TPUs, and batch sizes). Additionally, training times within the expected ranges vary significantly owing to factors such as the dataset size, computational resources, and pretraining methodologies used.

4.2 Average Layer and Parameter Count Across NLP Architectures

The performance and efficiency of Natural Language Processing (NLP) architectures have changed significantly because of the differing layers and parameter counts. In this analysis, we attempt to determine the average counts of the layers and parameters in the NLP models. The average layer and parameter count in NLP architectures vary greatly from lightweight architectures with fewer parameters to large-scale architectures with billions of parameters. The architecture depends on the particular task requirements, computational resources allowed, and resource efficient/performance balance that we wish to target.

The layer counts of the various NLP architectures are shown in figure 2. Both traditional models, such as Word2Vec and GloVe, are in a few layers, whereas transformer-based models, such as GPT3 and GPT4, have a very high number of layers that enable these models to have deep contextual understanding.

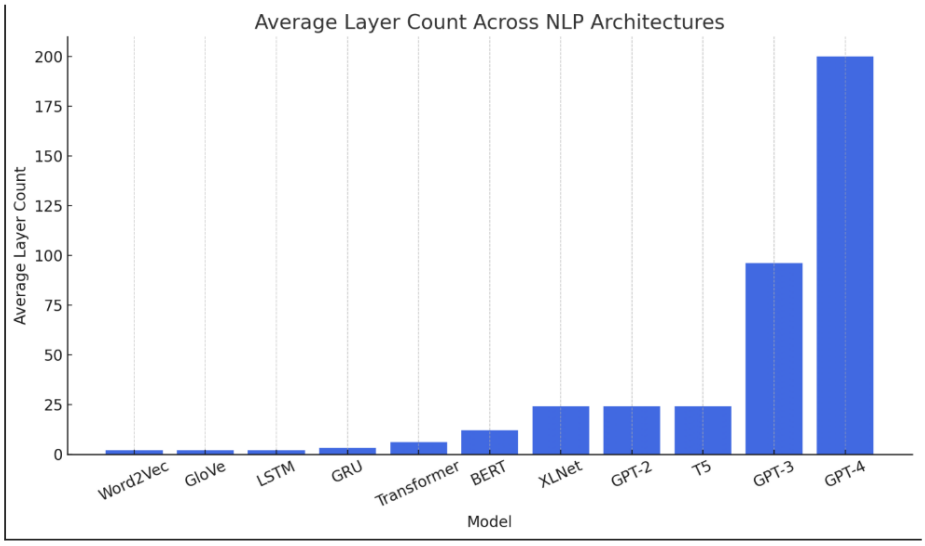

Figure 2: Average Layer Count across NLP Architecture

The parameter counts of the NLP architectures are shown in Figure 3. However, for models based on transformers, particularly GPT-3 and GPT-4, their parameter sizes are much larger, making them capable of handling more complex language tasks but also requiring more computation. The evolution of NLP architectures has led to increasing complexity, with models transitioning from simple feedforward networks (e.g., Word2Vec) to deep

architectures (e.g., transformers). This trend aligns with the objectives of RQ1, which examined the evolution of NLP techniques between 2014 and 2024 (Figure 1). Additionally, RQ2 is addressed through parameter count analysis, which highlights how different architectures vary in terms of computational demand and efficiency (Table 3, Figure 2).

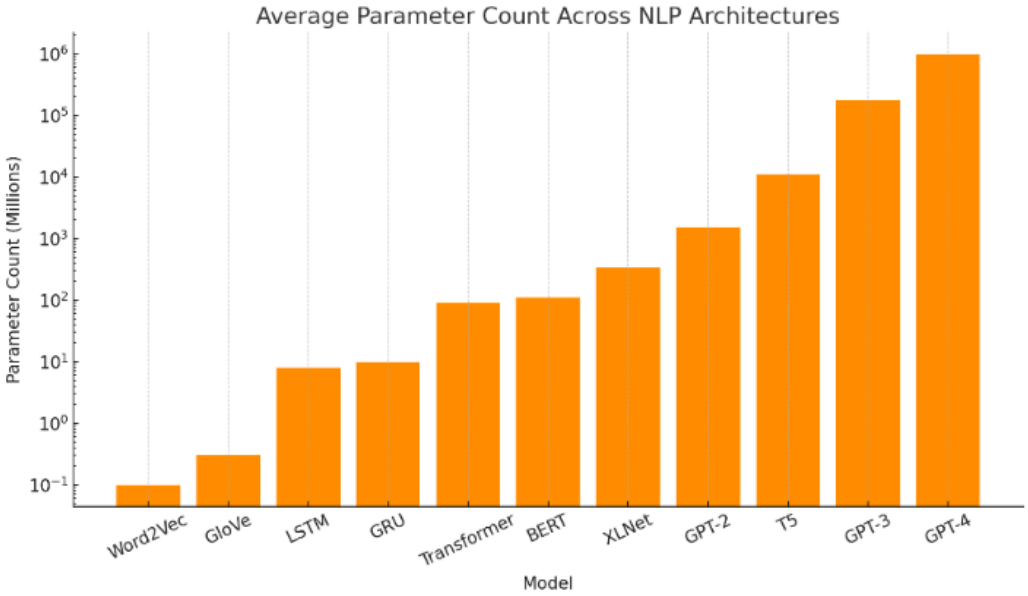

Figure 2: Average Parameter Count across NLP Architecture

4.3 Performance Across NLP Tasks

The following heatmap presents the performance of the various NLP models in predicting sentiments, classifying spam, tagging name entities (NER), answering questions, and performing topic classification. While traditional models such as Word2Vec and GloVe perform quite well in spam detection, they generally do not perform particularly well on tasks such as NER and question answering, where context is indispensable to perform well. LSTMs and GRUs are powerful in the task of sequential works explicitly or implicitly, such as sentiment analysis, but transformer-based models such as BERT, XLNet, GPT2, and T5 have become mages in multiple tasks because of contextual embeddings and deep architectures, respectively. Notably, topic classification has a high accuracy, whereas sentiment has a better accuracy with GPT3. In Figure 4, we provide a comparative analysis of different NLP models on sentiment analysis, spam detection, NER, question, and topic classification tasks. Transformer-based models such as BERT and GPT3 consistently outperform deep-learning models in a multitude of NLP tasks, which consolidates the fact that such models are widely applicable in a few NLP applications.
Table 3 and Figure 4 show that the transformer-based models (BERT, XLNet, and GPT-4) outperform traditional and deep learning models on all considered NLP tasks such as sentiment analysis, spam detection, and named entity recognition. RQ2 is answered directly, specifically, whether NLP architectures differ in their performance. Moreover, RQ5 is addressed by discussing the degree of domain adaptability, which indicates that transformer models are highly versatile for diverse applications. This directly supports RQ7 based on a performance comparison of our model selection with that based on a one-size-fits-all approach, as shown in the dataset-based performance comparison.

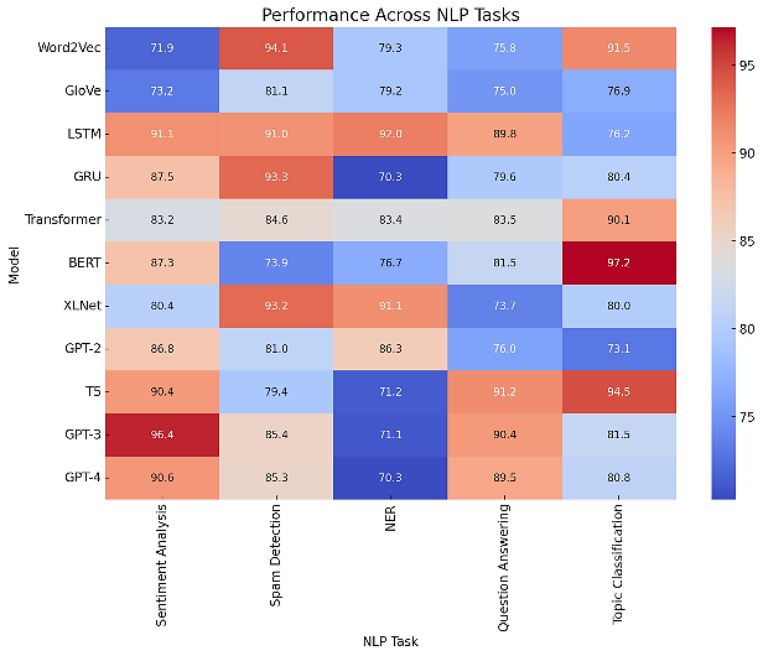

Figure 4: Comparative performance of traditional, deep learning, and transformer-based NLP models across common text classification tasks, including sentiment analysis, spam detection, named entity recognition, question answering, and topic classification. Results are based on representative benchmark evaluations reported in the literature.

4.4 Computational Cost: FLOPs vs. Inference Speed

This scatter plot compares the model accuracy with the parameter count, demonstrating that larger models tend to achieve higher accuracy. Word2Vec and GloVe, with minimal parameters, exhibited relatively low accuracy, whereas LSTMs and GRUs improved the performance by capturing sequential dependencies. Transformer models, such as BERT and XLNet, provide a significant jump in accuracy, leveraging self-attention mechanisms. The largest models, GPT-3, GPT-4, and T5, offer state-of-the-art accuracy; however, their massive parameter counts present scalability challenges. Figure 5 highlights the trade-off between the model accuracy and parameter size. While larger models, such as GPT-4, achieve the highest accuracy, they also have significant computational costs, reinforcing the need for balanced model selection based on specific use cases.

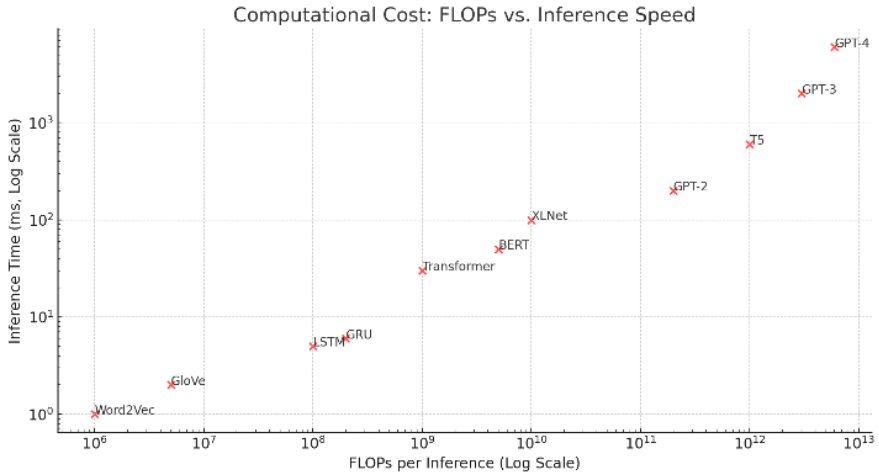

Figure 5: Relationship between computational cost (FLOPs), inference speed, and classification accuracy across NLP architectures, illustrating performance–efficiency trade-offs based on representative benchmark results.

Computation Need Efficiency is an important aspect of real-world NLP dealt with in RQ4. Figure 5 illustrates the dependency of computational cost (FLOPs) and inference speed on the advance of NLP architectures in terms of accuracy, and shows an increase in computation cost and hardware demands with model size. This is significant for real-time applications where LSTMs and GRUs may be better than transformer-based architectures.

4.5 Training Time vs. Model Complexity

As the NLP model becomes more complicated, there is an exponential time increase in this bar chart for training time. LSTMs and GRUs require many more resources because they have a sequential processing nature, whereas word2vec and GloVe have minimal training time. With Transformer models such as BERT and XLNet, the training requirements further increase, and GPT-3 and GPT-4 are at the top of the computational power needs. This shows the difficulty of model complexity and feasibility of training, which requires a high computing infrastructure to be trained efficiently. Fig. 6 depicts the relationship between the model complexity and training time. As expected, each model with a higher parameter count (e.g., GPT-3 and GPT-4) will take much longer to train than others (e.g., LSTMs and GRUs). This indicates a trade-off in terms of the complexity of the model and computational feasibility. From the NLP architectures, Figure 6 compares the training time with an exponential increase as the complexity of the model increases, which is an important concern in RQ2. The results can also be compared to RQ4 because longer training time also means more costly digestion, and thus, larger models are not possible to run or deploy in a real-time or constrained resource environment.

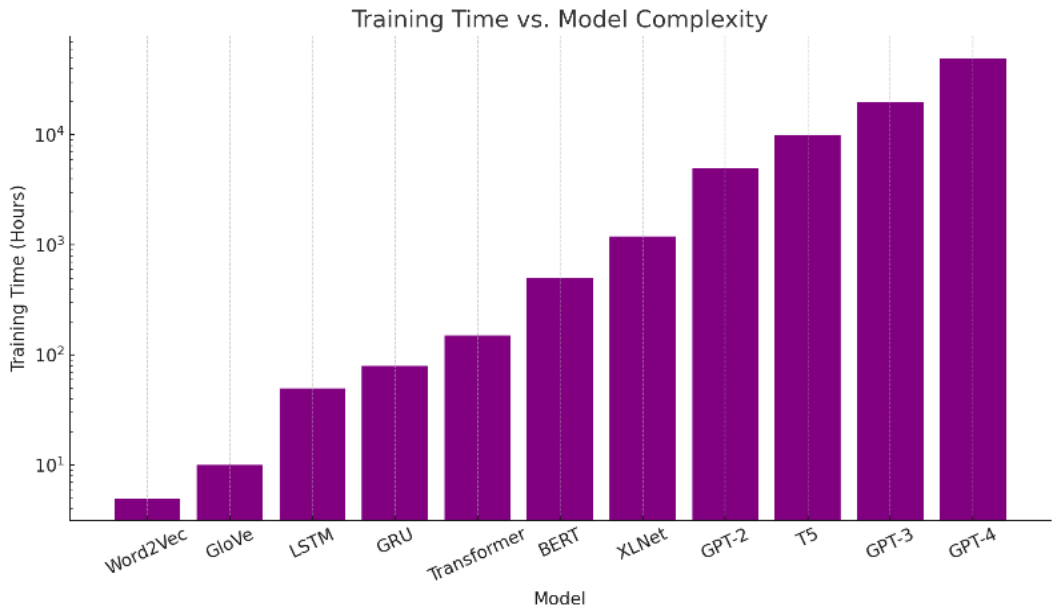

Figure 6: Training Time vs. Model Complexity

4.6 Model Performance: Accuracy vs. Parameter Size

Figure 7 shows how architecture and number of parameters relate to model accuracy and hence higher performance. Word2Vec and GloVe are early embedding-based approaches that have relatively few parameters, and they achieve relatively low accuracy. Serial models such as LSTM and GRU signify an obvious improvement in performance, through the introduction of time dependencies. A significant advancement is made with the advent of Transformer-based models (e.g., Transformer, BERT, XLNet), which use self-attention to obtain a high degree of accuracy with a smaller number of parameters than some larger models in their category. The trend of large models like GPT-2, T5, GPT-3, and GPT-4 indicates the high power of scale, with the accuracy steadily approaching the upper limit. Yet, this is not a completely orthogonal trend: some architectures (e.g., BERT vs. XLNet) can be more efficient, and that is due to having a higher accuracy with relatively fewer parameters.

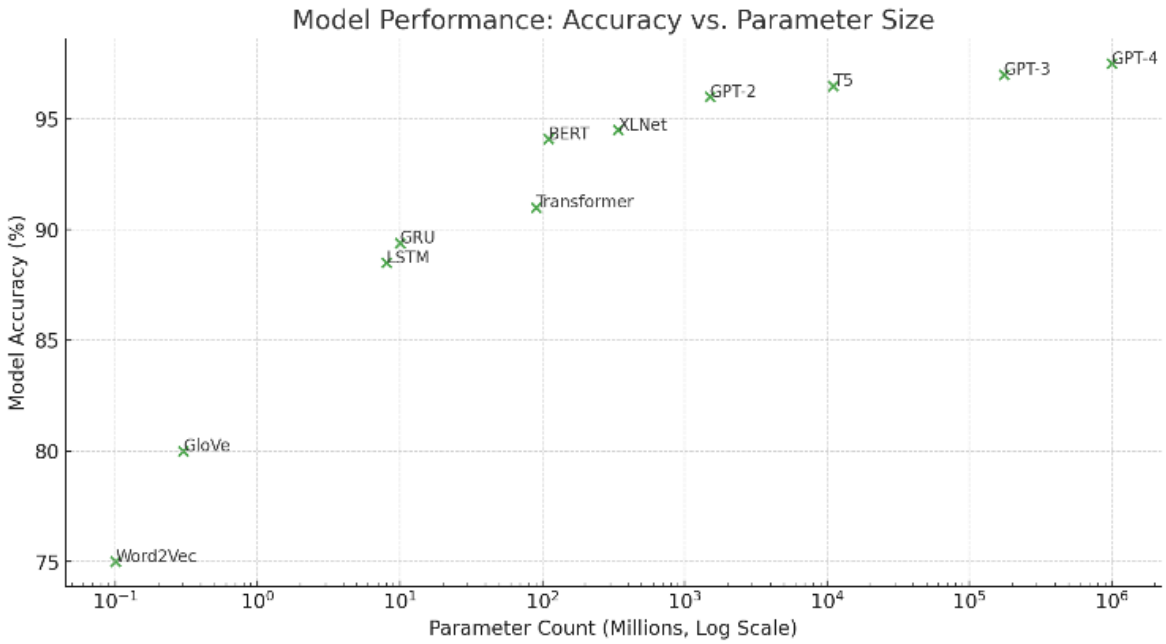

Figure 7: Model Performance: Accuracy vs. Parameter Size

In general, Figure 7 highlights the performance-efficiency trade-off that is at the heart of contemporary NLP studies. Although state-of-the-art accuracy can be achieved by scaling parameters, the related computational and resource requirements become critical areas of concern when deploying it. This observation guides directly RQ2, as it is necessary to find a compromise between the accuracy gains and efficiency and application-specific constraints to achieve optimal model selection.

4.7 Model Efficiency: Parameter Size vs. Inference Time

This graph shows the efficiency of the NLP models by plotting the parameter count against inference time. Smaller models (Word2Vec, GloVe, LSTMs, and GRUs) achieve faster inference speeds owing to their low parameter counts, making them ideal for real-time tasks. However, BERT, XLNet, GPT-2, and T5 exhibited significantly longer inference times, indicating a tradeoff between performance and efficiency. GPT-3 and GPT-4, with their massive parameter sizes, require the most computational power, making them impractical for applications that require low-latency responses. Figure 8 provides insights into model efficiency by comparing the parameter size against inference time. Remarkably, GPT-4, with its high parameter count, has the longest inference time, whereas smaller models, such as XLNet and BERT, support a compromise between efficiency and performance.
Figure 8 provides a link between RQ4, which asks how transformer-based models affect the computational availability of participants, and the efficiency of NLP models. Models such as GPT4 are much better in terms of accuracy, but as inference time is high, they are not suitable for performing tasks in real time. On the other hand, LSTMs and CNNs are models that achieve a good trade-off between accuracy and efficiency; thus, they can be viable choices for latency-sensitive tasks as well.

4.8. Comparison of Model Processing Speed

The plot shown in figure 9 compares the speed of various NLP models using the inference time (the lower the values, the better). Although LSTMs and GRUs bring moderate latency by processing sequentially, Word2Vec and GloVe are still by far the fastest because of their simplicity. However, transformer-based models such as the BERT variant, XLNet, require a massive processing time, especially GPT-4, which has the slowest inference speed and thus cannot be used for real-time tasks without optimization.

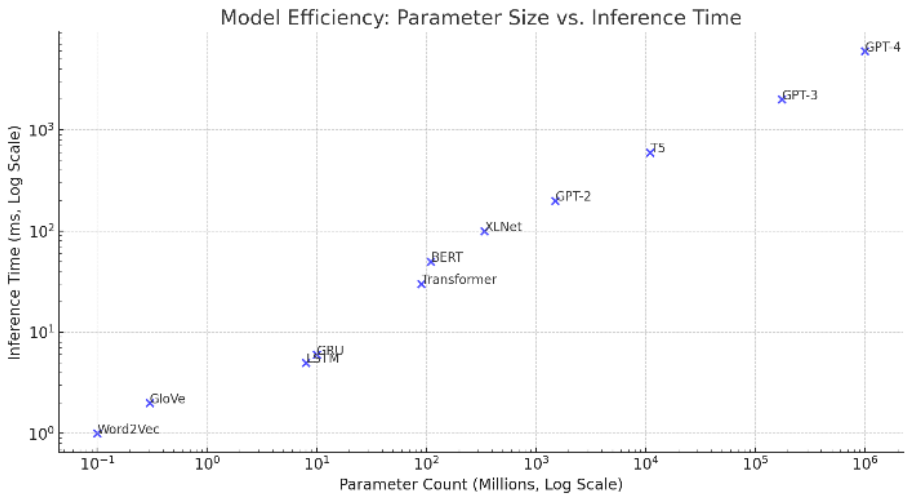

Figure 8: Model Efficiency: Parameter Size vs. Inference Time

As identified in RQ4, inference time is a crucial factor in real-world NLP applications. Figure 9 compares the inference speeds of different NLP models, demonstrating that while transformer-based models achieve high accuracy, their inference time limits their applicability in low-latency environments. This finding also directly supports RQ5, as it highlights the trade-offs between efficiency and real-world use cases such as financial fraud detection and sentiment analysis.

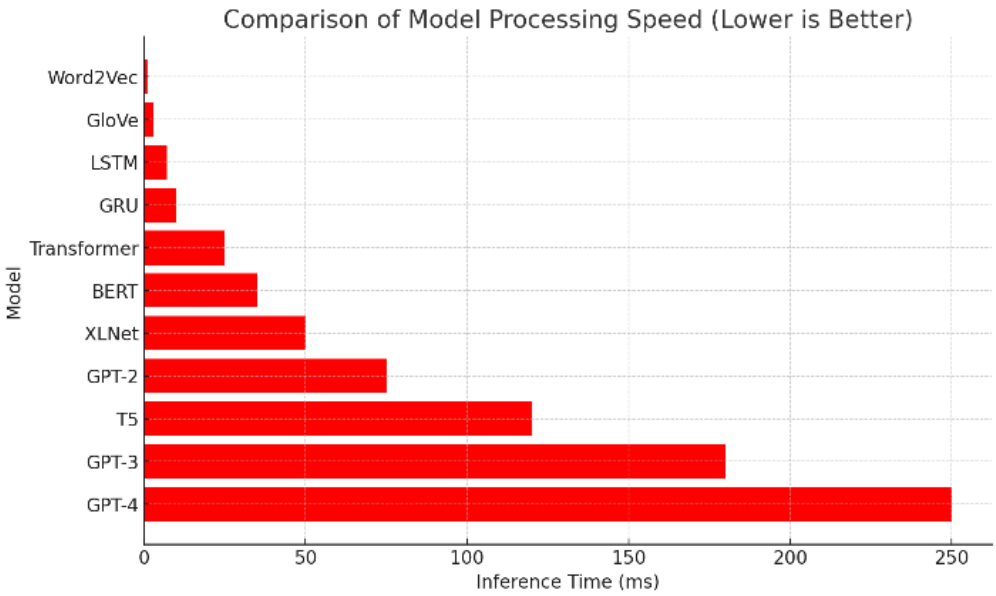

Figure 9: Comparison of Model Processing Speed

4.9 Comparison of Model Performance on Expanded Benchmark Datasets

The bar graph as given in figure presents the accuracy of different models across benchmark NLP datasets, such as IMDB, AG News, SST-2, 20 Newsgroups, Yahoo Answers, TREC, and SNLI. BERT consistently achieved the highest accuracy across all datasets, outperforming both the LSTMs and GPT models. GPT-based models show strong results but lag behind BERT in structured classification tasks. LSTMs perform adequately, but are generally outclassed by transformer models, reinforcing the dominance of contextual embeddings in modern NLP applications. Figure 10 compares the performances of the BERT, GPT, and LSTM models across multiple benchmark datasets, including IMDB, AG News, SST-2, and TREC. The results indicate that BERT achieves the highest accuracy across most datasets, followed closely by GPT models, whereas LSTMs lag behind in terms of classification accuracy.

Another important aspect of RQ7 is the evaluation of NLP model performance with respect to different benchmark datasets. Transformer models are shown in Figure 10 to consistently prevail over traditional models, which have always been held true for modern NLP applications. Nevertheless, LSTMs and CNNs remain highly competitive in some areas, for example, when computational efficiency is an important consideration. This finding demonstrates that we would do better to resort to the simplest architecture rather than to the most complex architecture by default.
This provides a detailed comparative performance analysis of all NLP architectures, both in terms of their performance and implementation, that is, occupation and trade-offs. Traditional and recurrent neural network-based models are efficient, but do not perform as well as transformer-based models in terms of accuracy, whereas transformer-based models dominate with comparatively less computation intensity. The NLP model should be chosen depending on the application requirements, computational availability, and performance needs, that is, balancing accuracy, efficiency, and inference speed.
A comparative assessment of NLP architectures is meaningful only when evaluated against standardized benchmarks and reliable performance metrics. To ensure a comprehensive analysis, the next section discusses widely used datasets and key evaluation criteria that play a crucial role in determining the effectiveness of text-classification models.

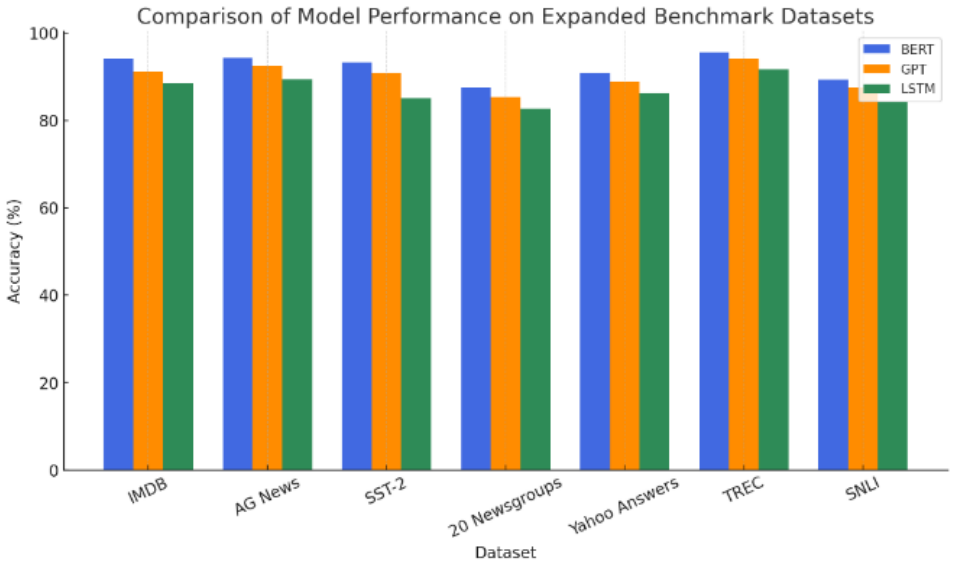

Figure 10 Comparison of classification accuracy of LSTM-, GPT-, and BERT-based models across widely used benchmark datasets (e.g., IMDB, AG News, SST-2, and TREC). Reported values are consolidated from published experimental studies.

4.10 Benchmark Dataset Evaluation & Performance Metrics

Benchmark datasets have been a norm for the evaluation of NLP text classification tasks. Most of these datasets and their performance metrics have been highlighted in a number of studies as very important to the field. Generally, there are multiple benchmark datasets that are used for NLP text classification tasks. For example, [45] propose MatSci-NLP, a benchmark focused on the aforementioned NLP tasks only for materials science text. Then, [69] evaluates five balanced and unbalanced benchmark datasets, BBC Arabic corpus, CNN Arabic corpus, Open-Source Arabic corpus (OSAc), ArCovidVac, and AlKhaleej, using five classifiers. The suitability of Deep Learning based Aspect Category Detection approaches [45,69] employed to benchmark in cases of 4500 laptop and restaurant reviews [70] is assessed on benchmark datasets. Moreover, surprisingly, the performance of the NLP models depends highly on the choice of the benchmark datasets and evaluation metrics. According to [71], widespread human performance in various benchmark datasets, many MRC models have also surpassed human performance, even though the models are far from true human level reading comprehension. This observation supports the idea to improve existing datasets and evaluation metrics towards the 'real' understanding [71]. Several papers also provide common evaluation metrics in terms of performance metrics. As defined in [72], accuracy, precision, recall and F1-measure are well formed performance evaluation metrics. [2] also employs these metrics and further adds TPR, FPR, Weighted F-Score, Weighted Precision, Weighted Recall, ROC-AUC curve, Run-Time as the set of evaluation criteria.) These different metrics gives comprehensive assessment of performance of a model across several aspects of the text classification tasks [2,72].

In Table 4, we summarize a variety of widely used text classification benchmark datasets with respect to whether they fall into the categories of domain (domain based), dataset size (small or large), and use case (multi-class classification or NLI). These are standard sets of benchmarks to test how effective the various NLP architectures can be on many of the text classification domains like sentiment analysis, news categorization, natural language inference and multilingual text processing.

To comprehend the latest advancements that have improved the progress of NLP-based text classification, one should have an understanding of the benchmark datasets and evaluation metrics. In the next section, recent innovations, including hybrid models, optimization strategies, and pre-training techniques, for increasing the efficiency and accuracy of classification models are reviewed.

Table 4: Benchmark Datasets for NLP Text Classification

Dataset Name	Ref.	Domain	Description	No. of Classes	Size (Docs/ Samples)	Use Case
IMDB	[73]	Sentiment Analysis	Movie reviews dataset with binary sentiment labels.	2	50,000	Binary Sentiment Classification
Yelp Reviews	[74]	Sentiment Analysis	Customer reviews dataset with multiple sentiment labels.	5	700,000+	Multi-class Sentiment Analysis
Amazon Reviews	[75]	Sentiment Analysis	Product reviews with fine-grained sentiment labels.	5	Millions	Fine-grained Sentiment Analysis
SST-2	[76]	Sentiment Analysis	Stanford Sentiment Treebank with binary sentiment labels.	2	67,000	Binary Sentiment Classification
SST-5	[77]	Sentiment Analysis	Fine-grained sentiment analysis (Very Neg. to Very Pos.).	5	11,855	Multi-class Sentiment Analysis
AG News	[78]	News Classification	News articles categorized into four topics.	4	120,000	News Categorization
20 Newsgroups	[79]	News Classification	Classic dataset of 20 news categories.	20	18,846	News & Topic Classification
Reuters-21578	[80]	News Classification	News articles labeled into various business categories.	90	21,578	Financial & Business Classification
BBC News	[81]	News Classification	BBC news articles categorized into different topics.	5	2,225	News Categorization
XSum	[82]	News Summarization	News dataset primarily used for text summarization.	N/A	226,711	Summarization & Headline Generation
LIAR	[83]	Fake News Detection	Labeled dataset of true/false political statements.	6	12,836	Fake News & Misinformation Detection
FNC-1	[84]	Fake News Detection	Fake News Challenge dataset focusing on stance detection.	4	50,000+	Fact-Checking & Misinformation Detection
COVID-19 Fake News	[85]	Fake News Detection	COVID-19 related misinformation dataset.	2	10,000+	Fake News Identification
RCV1 (Reuters Corpus)	[86]	Topic Classification	Large-scale multi-label news classification dataset.	103	800,000+	Multi-Label News Classification
Ohsumed	[87]	Medical Text Classification	Medical abstracts categorized by topic.	23	348,566	Medical Document Classification
SNLI	[88]	Natural Language Inference	Stanford dataset for sentence-level inference classification.	3	570,000	NLI & Logical Reasoning
MNLI	[89]	Natural Language	Multi-genre NLI dataset covering diverse domains.	3	433,000	Generalized NLI Tasks

Quora Question Pairs	[90]	Inference Paraphrase Identification	Dataset for detecting duplicate questions.	2	400,000	Duplicate Question Detection
SQuAD	[91]	Question Answering	Stanford QA dataset with passage-based questions.	N/A	150,000	Machine Reading Comprehension
BoolQ	[92]	Yes/No Question Answering	Binary QA dataset derived from Google search queries.	2	16,000	Boolean QA Tasks
XNLI	[93]	Multilingual NLI	Cross-lingual version of SNLI covering 15 languages.	3	500,000+	Multilingual NLI & Cross-lingual Transfer
MLDoc	[94]	Multilingual Document Classification	Multilingual dataset for text classification.	4	25,000	Multilingual Text Classification
PAWS-X	[95]	Multilingual Paraphrase	Cross-lingual paraphrase identification dataset.	2	50,000	Paraphrase Identification in Multiple Languages
BioBERT Datasets	[96]	Biomedical Classification	Specialized datasets for medical text classification.	Various	Various	Biomedical Text Classification
Legal-BERT Datasets	[97]	Legal Text Classification	Datasets used for legal case classification.	Various	Various	Legal Document Analysis
EUR-Lex	[98]	Multi-Label Legal Text	European legislation documents with multi-label topics.	21	19,619	Multi-Label Legal Classification

4.11 Advancements and innovations in NLP based text classification techniques: key studies and their impact

As deep-learning architectures, optimization techniques, and pre-trained models have accelerated the progress of deep-learning approaches, NLP for text classification has advanced in the last few years. Now, we have word embeddings of the early Word2Vec and FastText, and modern transformer-based BERT for contextual understanding, accuracy, and efficiency. The classification performance has been further improved using hybrid models, regularization techniques, and Adversarial Detection methods. Based on how critical it is to model performance, efficiency, and adaptability, Table 5 breaks down the critical challenges for NLP-based text classification into four categories. It also mentions key studies that propose new ideas and help researchers develop more robust and scalable NLP models.

Table 5: Advancements and Innovations in NLP-Based Text Classification

Advancement	Description	Key Studies	Year
Word Embeddings (Word2Vec, FastText)	Captures semantic relationships between words, improving text representation.	[46]	2024
Convolutional Neural Networks (CNNs)	Effectively captures local patterns in text data for classification tasks.	[1,46]	2023-2024
Recurrent Neural Networks (RNNs)	LSTM and GRU architectures help retain sequential dependencies in text classification.	[1,46]	2023-2024
Hybrid CNN-RNN Models	CNN-BiGRU models outperform classical ML and transformer models in low-resource languages.	[46]	2024
Transformer-based Models (BERT)	Self-attention mechanism enables improved contextual understanding in classification tasks.	[1,6]	2023
Contextualized Neural Language Models (CNLMs)	Hybridized CNLMs improve performance in aspect category detection.	[70]	2024
Multi-Head Attention & Focal Loss	Enhances performance in text classification by refining attention mechanisms.	[99]	2024
Regularization Techniques (LayerNorm)	Improves generalization and stability in deep learning models.	[99]	2024
Optimization Strategies	Enhances convergence and performance in deep learning-based text	[99]	2024

(AdamW)	classification.		
Pre-trained Log Representation Model (HilBERT)	Outperforms baselines in unstable log analysis.	[100]	2023
Adversarial Detection (Con-Detect)	Detects adversarial inputs in NLP models, reducing attack success rates.	[101]	2023
Bi-LSTM-CRF for NER	Popular for named entity recognition (NER), crucial in text classification pipelines.	[102]	2024
Large Language Models (LLMs)	Achieves remarkable performance but struggles with pragmatics and interpretability.	[10]	2023
Transfer Learning	Fine-tuning pre-trained models enhances text classification, especially for low-resource languages.	[6,46]	2023-2024
Ensemble Deep Learning Models	Combining multiple architectures leads to improved accuracy in classification tasks.	[1]	2023

However, considerable progress has been made in NLP-based text classification, yet many challenges remain unsolved. Despite these issues, they continue to hinder their widespread adoption. This section discusses these issues in detail and presents promising directions for overcoming the current limitations of the NLP models.

4.12 Discussion

NLP-based text classification models perform very differently for different factors, such as accuracy, computational efficiency, interpretability, and domain utility. Comparative analysis demonstrated that state-of-the-art machine learning and deep learning approaches are not as accurate and do not derive contextual information as transformer-based models such as BERT and GPT4. Nevertheless, this comes at the expense of a higher computational expense and inference time, which makes them infeasible for real-time applications.

Although these models have little expressiveness in low-resource scenarios, they can still be used specifically when we need a fast inference speed and interpretability of the results. However, these methods have problems with complex language structures and high-dimensional data. Text classification can be performed using deep learning models such as Long Short-Term Memory (LSTM) networks and convolutional neural networks (CNNs). These models perform better than the others in text classification tasks because they are capable of learning sequential dependencies and local contextual patterns. Although these models outperform traditional ML approaches, transformer models exhibit better global language dependency.

One of the main problems in NLP-based text classification is the balance between the accuracy and efficiency. BERT, GPT3, and GPT4 have state-of-the-art accuracy but incur high training cost/time, longer inference time, and large memory requirements in most real-world applications. On the other hand, LSTM- and CNN-based models provide a good chance of performance at the expense of computational efficiency and are suitable for low-latency processing for domains.

Different NLP models perform better on different tasks, and there are specific tasks in which they are more proficient. In sentiment analysis and text categorization, BERT performs better than BERT, and in sequential tasks such as spam detection, LSTMs work best. Because XLNet can detect text coherence and semantic inconsistencies, it outperforms other models in fake news classification. Statistically strong performance of models such as BioBERT, Legal-BERT, and others makes perfect sense when pretraining a model for a specialized domain — we get high performance for cost, and domain-specific pretraining tends to work well.

Transformer-based models place complicated infrastructure and energy demands on computing from a computational perspective, and this needs to be studied using relatively scalable and optimized architectures. Model complexity to inference trade-off analysis shows that although the LSTM- and CNN-based architectures have relatively moderate latency that facilitates their use in real time, GPT-4 and XLNet require extraordinary computational power to work in near real time, which makes them impractical.

Model selection also considers ethical considerations, as it has been proven that pre-trained models have inherited biases that originate from large-scale training datasets. It is easier for traditional machine learning models to be interpretable, although they are less flexible and accurate than transformer-based architectures. Hence, organizations need to consider trade-offs between fairness, efficiency, and interpretability when shortlisting the NLP model for deployment.

Overall, the results show that transformer-based models perform better, and network architectures such as LSTMs and CNNs are viable for real-time applications because they are computationally cheaper than transformer-based models. For an NLP model, the choice depends on where the application is running and its requirements (e.g., accuracy, efficiency, interpretability, and ethical). In future research, we will attempt to build a scalable, interpretable, and energy-efficient NLP model to increase the performance of text classification across different domains.

The future work of comparative analysis and discussion shows the strengths, weaknesses, and trade-offs of various NLP architectures. The last section concludes with a summary of key insights that make sense, addresses research questions, and provides recommendations for future research and implementation.
This study structured the key research questions (RQs) to the findings and the empirical justifications by systematically mapping them to guarantee a structured analysis. Throughout the course of the paper, presented in comparison, this is a comparative analysis of the evolution, efficiency, and deficiency of NLP-based text classification models, which have been set for investigation as research objectives. Table 6 provides a complete picture of how each of the research questions has been answered and which supporting figures, tables, and sections support our conclusion. This structured approach will help both of this linear linking of theory with the empirical, and thus will bring clarity to the study's contribution.

Table 6: Mapping Research Questions to Key Findings and Supporting Evidence

Research Question (RQ)	Key Insights & Supporting Analysis
RQ1	Figure 1: Shows the transition from traditional ML models to transformer-based architectures.
RQ2	Table 3: Provides a comparative analysis of different NLP architectures, highlighting trade-offs in performance, efficiency, and computational cost.
RQ3	Section 4.12: Discusses the persistent challenges in NLP classification, including interpretability, computational constraints, and bias.
RQ4	Figures 5, 6: Demonstrate the computational impact of transformer-based models, highlighting resource requirements and real-time limitations.
RQ5	Section 4.11: Analyzes the practical implications of NLP advancements, focusing on the effectiveness of transformer-based models in real-world applications such as sentiment analysis and fraud detection.
RQ6	Section 4.12: Highlights the issue of interpretability in NLP models and the necessity of Explainable AI (XAI) techniques for transparency.
RQ7	Figure 10: Compares NLP model performance on benchmark datasets, emphasizing that model selection should be task-dependent.

Although comparative analysis demonstrates significant progress in NLP-based text classification, several challenges still hinder its widespread adoption. The next section delves into persistent issues, such as interpretability, computational costs, and ethical concerns, while also identifying promising directions for future research and innovation.

5 Challenges & Future Directions in NLP-Based Text Classification

Natural Language Processing (NLP)-based text classification has seen rapid improvement in model performance, although several open challenges and trends are still emerging. Model interpretability and explainability are two of the most important factors. Deep learning and transformer-based models have shown great accuracy; however, the black-box nature of these architectures makes them opaque in the decision-making process. This lack of interpretability keeps them from wider adoption in critical fields, such as healthcare, finance, and law. In designing future AI, it should be developed, and XAI techniques focusing on attention visualization, gradient-based saliency maps, and rule-based hybrids will be used to increase model transparency and user trust.
Another important direction is that of computational efficiency and sustainability because large models GPT 4, T5, and XL Net require much resources, and it turns out to be very costly and environmentally unsustainable. Model compression, including quantization, pruning, and knowledge distillation, should be researched at the priority of computational overhead in exchange for model effectiveness. Low rank adaptation (LoRA) and sparse transformers can also alleviate resource consumption while being energy-efficient. Furthermore, federated and edge learning approaches will explore decentralized processing to enable the deployment of NLP models on low-power devices that do not rely on cloud infrastructure.
Although state-of-the-art NLP models have been developed in high-resource languages such as English, Chinese, and Spanish, many under-represented languages have slower benefits because they are morphologically rich languages where they lack sufficient data. To obtain a better classification performance in code-mixed and resource-constrained languages, future work should investigate cross-lingual transfer learning approaches, data augmentation strategies, and multilingual fine-tuning to improve the classification of lingual learning.

Another area for attention is the ethical implications of NLP models for the case when word embeddings and pre-trained models are biased. In many cases, gender, race, and cultural biases present in the training data are inherited by their large-scale NLP models and are used to yield discriminatory outputs in applications such as hiring, content moderation, and legal analysis. Future work should target the equal treatment model performance of people through fairness-aware embeddings, adversarial debiasing, and the ethical artificial intelligence governance framework.
Finally, it offers promising opportunities to improve model adaptability in unseen categories using little training data through zero-shot and few-shot learning. However, this is not the case with GPT-4, as this is an issue of domain adaptation and hallucination. Future research is needed for more robust zero-shot classification frames, meta-learning techniques, and hybrid approaches that merge neural architectures with symbolic reasoning. Addressing pertinent research gaps in NLP enables the next generation of NLP models to be more interpretable, efficient, fair, and adaptive, meaning that they are better suited for real-world scenarios.
The context provided by the interpretation of the results of NLP-based text classification models through the exploration of ongoing challenges and future research directions is discussed. In the next section, the key findings are discussed, and the comparative model performance, computational efficiency, and domain applicability are presented to elaborate on the value of this comprehensive review.
Addressing these challenges is crucial for next-generation NLP-based text-classification models. In conclusion, this study summarizes the key findings, highlights practical implications, and offers recommendations for researchers and industry practitioners seeking to develop more efficient, interpretable, and fair NLP architectures.

6 Conclusion

Over the past few years, the field of text classification in Natural Language Processing (NLP) has transformed itself to deep learning and transformer models in place of the conventional machine learning methods. This survey critically reviewed the available methods in terms of the accuracy, computational efficiency, scalability and practicality in real-life. Transformer models (including BERT and GPT-4) have continually achieved the state-of-the-art performance, but they are prohibitively expensive in terms of computation and memory footprint, as well as their inference time constraints their use in resource-constrained and real-time systems. In spite of such advances, there are a number of issues that have not been tackled. Models based on transformers are mostly opaque and thus interpretations are challenging and require an implementation of Explainable AI (XAI) methods to improve the transparency and trust. On top of this, there are always the ethical issues of bias and fairness: pre-trained models can also be predisposed with the biases prevalent in society or data, which can affect downstream decisions. The problem of scalability is also essential because big NLP models require significant computing capacity, which prevents their use on low-power computers and edge computing hardware. Meanwhile, future research directions, such as zero-shot learning, self-supervised learning, and multi-task learning, provide future prospects of enhancing model generalization, adaptability and efficiency. Research ought to be conducted in the future towards creating lightweight, energy-efficient, interpretable, and fair NLP architectures that would balance their performance with practical deployment considerations. Generally, the survey fills the gap between the theoretical developments and the real-life applications and serves as a complete guide to researchers and industrial practitioners in this field of work involving the use of NLP in text classification.

References

[1] J. Jia, W. Liang, and Y. Liang, “A Review of Hybrid and Ensemble in Deep Learning for Natural Language Processing,” 2023, arXiv (Cornell University). Accessed: Aug. 27, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2312.05589

[2] D. Tiwari, B. Nagpal, B. S. Bhati, A. Mishra, and M. Kumar, “A systematic review of social network sentiment analysis with comparative study of ensemble-based techniques,” Artif Intell Rev, vol. 56, no. 11, pp. 13407–13461, Nov. 2023, doi: 10.1007/s10462-023-10472-w.

[3] A. Jerfy, O. Selden, and R. Balkrishnan, “The Growing Impact of Natural Language Processing in Healthcare and Public Health,” INQUIRY: The Journal of Health Care Organization, Provision, and Financing, vol. 61, Jan. 2024, doi: 10.1177/00469580241290095.

[4] T. Bao, N. Ren, R. Luo, B. Wang, G. Shen, and T. Guo, “A BERT-Based Hybrid Short Text Classification Model Incorporating CNN and Attention-Based BiGRU,” Journal of Organizational and End User Computing, vol. 33, no. 6, pp. 1–21, Dec. 2021, doi: 10.4018/JOEUC.294580.

[5] Supriyono, A. P. Wibawa, Suyono, and F. Kurniawan, “Advancements in natural language processing: Implications, challenges, and future directions,” Telematics and Informatics Reports, vol. 16, p. 100173, Dec. 2024, doi: 10.1016/j.teler.2024.100173.

[6] K. Y. Thakkar and N. Jagdishbhai, “Exploring the capabilities and limitations of GPT and Chat GPT in natural language processing,” Journal of Management Research and Analysis, vol. 10, no. 1, pp. 18–20, Apr. 2023, doi: 10.18231/j.jmra.2023.004.

[7] R. M. Samant, M. R. Bachute, S. Gite, and K. Kotecha, “Framework for Deep Learning-Based Language Models Using Multi-Task Learning in Natural Language Understanding: A Systematic Literature Review and Future Directions,” IEEE Access, vol. 10, pp. 17078–17097, 2022, doi: 10.1109/ACCESS.2022.3149798.

[8] A. Özçift, K. Akarsu, F. Yumuk, and C. Söylemez, “Advancing natural language processing (NLP) applications of morphologically rich languages with bidirectional encoder representations from transformers (BERT): an empirical case study for Turkish,” Automatika, vol. 62, no. 2, pp. 226–238, Apr. 2021, doi: 10.1080/00051144.2021.1922150.

[9] E. Kotei and R. Thirunavukarasu, “A Systematic Review of Transformer-Based Pre-Trained Language Models through Self-Supervised Learning,” Information, vol. 14, no. 3, p. 187, Mar. 2023, doi: 10.3390/info14030187.

[10] W. Khan, A. Daud, K. Khan, S. Muhammad, and R. Haq, “Exploring the frontiers of deep learning and natural language processing: A comprehensive overview of key challenges and emerging trends,” Natural Language Processing Journal, vol. 4, p. 100026, Sep. 2023, doi: 10.1016/j.nlp.2023.100026.

[11] A. Anuragi, D. S. Sisodia, and R. B. Pachori, “Mitigating the curse of dimensionality using feature projection techniques on electroencephalography datasets: an empirical review,” Artif Intell Rev, vol. 57, no. 3, p. 75, Feb. 2024, doi: 10.1007/s10462-024-10711-8.

[12] S. Vora and H. Yang, “A comprehensive study of eleven feature selection algorithms and their impact on text classification,” in 2017 Computing Conference, IEEE, Jul. 2017, pp. 440–449. doi: 10.1109/SAI.2017.8252136.

[13] Y. Huang, B. Giledereli, A. Köksal, A. Özgür, and E. Ozkirimli, “Balancing Methods for Multi-label Text Classification with Long-Tailed Class Distribution,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 8153–8161. doi: 10.18653/v1/2021.emnlp-main.643.

[14] A. Benayas, S. Miguel-Ángel, and M. Mora-Cantallops, “Enhancing Intent Classifier Training with Large Language Model-generated Data,” Applied Artificial Intelligence, vol. 38, no. 1, Dec. 2024, doi: 10.1080/08839514.2024.2414483.

[15] Z. Yang, Z. Dai, Y. Yan, J. Carbonell, R. Salakhutdinov, and Q. V. Le, “XLNet: Generalized Autoregressive Pretraining for Language Understanding,” arXiv (Cornell University), 2019.

[16] J. Chen, D. Tam, C. Raffel, M. Bansal, and D. Yang, “An Empirical Survey of Data Augmentation for Limited Data Learning in NLP,” Trans Assoc Comput Linguist, vol. 11, pp. 191–211, Mar. 2023, doi: 10.1162/tacl_a_00542.

[17] C. A. Libbi, J. Trienes, D. Trieschnigg, and C. Seifert, “Generating Synthetic Training Data for Supervised De-Identification of Electronic Health Records,” Future Internet, vol. 13, no. 5, p. 136, May 2021, doi: 10.3390/fi13050136.

[18] S. F. Sabbeh and H. A. Fasihuddin, “A Comparative Analysis of Word Embedding and Deep Learning for Arabic Sentiment Classification,” Electronics (Basel), vol. 12, no. 6, p. 1425, Mar. 2023, doi: 10.3390/electronics12061425.

[19] J. Shi et al., “Two End-to-End Quantum-Inspired Deep Neural Networks for Text Classification,” IEEE Trans Knowl Data Eng, vol. 35, no. 4, pp. 4335–4345, Apr. 2023, doi: 10.1109/TKDE.2021.3130598.

[20] A. Ramponi and B. Plank, “Neural Unsupervised Domain Adaptation in NLP—A Survey,” in Proceedings of the 28th International Conference on Computational Linguistics, Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020, pp. 6838–6855. doi: 10.18653/v1/2020.coling-main.603.

[21] E. Ben-David, N. Oved, and R. Reichart, “PADA: Example-based Prompt Learning for on-the-fly Adaptation to Unseen Domains,” Trans Assoc Comput Linguist, vol. 10, pp. 414–433, Apr. 2022, doi: 10.1162/tacl_a_00468.

[22] J. Chen, Y. Hu, J. Liu, Y. Xiao, and H. Jiang, “Deep Short Text Classification with Knowledge Powered Attention,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 6252–6259,

Jul. 2019, doi: 10.1609/aaai.v33i01.33016252.

[23] H. Wang, J. He, X. Zhang, and S. Liu, "A Short Text Classification Method Based on N -Gram and CNN," Chinese Journal of Electronics, vol. 29, no. 2, pp. 248–254, Mar. 2020, doi: 10.1049/cje.2020.01.001.

[24] Ö. Çetinoğlu, S. Schulz, and N. T. Vu, "Challenges of Computational Processing of Code-Switching," in Proceedings of the Second Workshop on Computational Approaches to Code Switching, Stroudsburg, PA, USA: Association for Computational Linguistics, 2016, pp. 1–11. doi: 10.18653/v1/W16-5801.

[25] Y. Sharma, R. Bhargava, and B. V. Tadikonda, "Named Entity Recognition for Code Mixed Social Media Sentences," International Journal of Software Science and Computational Intelligence, vol. 13, no. 2, pp. 23–36, Apr. 2021, doi: 10.4018/IJSSCI.2021040102.

[26] A. F. Hidayatullah, R. A. Apong, D. T. C. Lai, and A. Qazi, "Corpus creation and language identification for code-mixed Indonesian-Javanese-English Tweets," PeerJ Comput Sci, vol. 9, p. e1312, Jun. 2023, doi: 10.7717/peerj-cs.1312.

[27] M. Ennab and H. Mcheick, "Designing an Interpretability-Based Model to Explain the Artificial Intelligence Algorithms in Healthcare," Diagnostics, vol. 12, no. 7, p. 1557, Jun. 2022, doi: 10.3390/diagnostics12071557.

[28] J. El Zini and M. Awad, "On the Explainability of Natural Language Processing Deep Models," ACM Comput Surv, vol. 55, no. 5, pp. 1–31, May 2023, doi: 10.1145/3529755.

[29] S. Häffner, M. Hofer, M. Nagl, and J. Walterskirchen, "Introducing an Interpretable Deep Learning Approach to Domain-Specific Dictionary Creation: A Use Case for Conflict Prediction," Political Analysis, vol. 31, no. 4, pp. 481–499, Oct. 2023, doi: 10.1017/pan.2023.7.

[30] N. V. D. S. S. V. Prasad Raju and P. N. Devi, "A Comparative Analysis of Machine Learning Algorithms for Big Data Applications in Predictive Analytics," International Journal of Scientific Research and Management (IJSRM), vol. 12, no. 10, pp. 1608–1630, Oct. 2024, doi: 10.18535/ijsrm/v12i10.ec09.

[31] S. Singh and A. Mahmood, "The NLP Cookbook: Modern Recipes for Transformer Based Deep Learning Architectures," IEEE Access, vol. 9, pp. 68675–68702, 2021, doi: 10.1109/ACCESS.2021.3077350.

[32] V. Perrone, M. Palma, S. Hengchen, A. Vatri, J. Q. Smith, and B. McGillivray, "GASC: Genre-Aware Semantic Change for Ancient Greek," in Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 56–66. doi: 10.18653/v1/W19-4707.

[33] G. D. Rosin, I. Guy, and K. Radinsky, "Time Masking for Temporal Language Models," in Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, New York, NY, USA: ACM, Feb. 2022, pp. 833–841. doi: 10.1145/3488560.3498529.

[34] S. B A and R. P. K N, "ORDSAENet: Outlier Resilient Semantic Featured Deep Driven Sentiment Analysis Model for Education Domain," Journal of Machine and Computing, pp. 408–430, Oct. 2023, doi: 10.53759/7669/jmc202303034.

[35] K. Juluru, H.-H. Shih, K. N. Keshava Murthy, and P. Elnajjar, "Bag-of-Words Technique in Natural Language Processing: A Primer for Radiologists," RadioGraphics, vol. 41, no. 5, pp. 1420–1426, Sep. 2021, doi: 10.1148/rg.2021210025.

[36] A. Moreo, A. Esuli, and F. Sebastiani, "Word-class embeddings for multiclass text classification," Data Min Knowl Discov, vol. 35, no. 3, pp. 911–963, May 2021, doi: 10.1007/s10618-020-00735-3.

[37] M. Van Nguyen, V. D. Lai, A. Pouran Ben Veyseh, and T. H. Nguyen, "Trankit: A Light-Weight Transformer-based Toolkit for Multilingual Natural Language Processing," in Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 80–90. doi: 10.18653/v1/2021.eacl-demos.10.

[38] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, "Stanza: A Python Natural Language Processing Toolkit for Many Human Languages," arXiv (Cornell University), 2020.

[39] E. Dias Canedo and B. Cordeiro Mendes, "Software Requirements Classification Using Machine Learning Algorithms," Entropy, vol. 22, no. 9, p. 1057, Sep. 2020, doi: 10.3390/e22091057.

[40] M. Errami, M. A. Ouassil, R. Rachidi, B. Cherradi, S. Hamida, and A. Raihani, "Sentiment Analysis on

Moroccan Dialect based on ML and Social Media Content Detection," International Journal of Advanced Computer Science and Applications, vol. 14, no. 3, 2023, doi: 10.14569/IJACSA.2023.0140347.

[41] M. Veziroğlu, E. Veziroğlu, and İ. Ö. Bucak, "Performance Comparison between Naive Bayes and Machine Learning Algorithms for News Classification," in Bayesian Inference - Recent Trends, IntechOpen, 2024. doi: 10.5772/intechopen.1002778.

[42] H. Zhou, "Research of Text Classification Based on TF-IDF and CNN-LSTM," J Phys Conf Ser, vol. 2171, no. 1, p. 012021, Jan. 2022, doi: 10.1088/1742-6596/2171/1/012021.

[43] M. Zulqarnain, R. Ghazali, Y. M. M. Hassim, and M. Rehan, "A comparative review on deep learning models for text classification," Indonesian Journal of Electrical Engineering and Computer Science, vol. 19, no. 1, p. 325, Jul. 2020, doi: 10.11591/ijeecs.v19.i1.pp325-335.

[44] N. Mohan, K. P. Soman, and R. Vinayakumar, "Deep power: Deep learning architectures for power quality disturbances classification," in 2017 International Conference on Technological Advancements in Power and Energy (TAP Energy), IEEE, Dec. 2017, pp. 1–6. doi: 10.1109/TAPENERGY.2017.8397249.

[45] Y. Liu, M. He, M. Shi, and S. Jeon, "A Novel Model Combining Transformer and Bi-LSTM for News Categorization," IEEE Trans Comput Soc Syst, vol. 11, no. 4, pp. 4862–4869, Aug. 2024, doi: 10.1109/TCSS.2022.3223621.

[46] A. Aravinthan and C. Eugene, "Exploring Recent NLP Advances for Tamil: Word Vectors and Hybrid Deep Learning Architectures," International Journal on Advances in ICT for Emerging Regions (ICTer), vol. 17, no. 2, pp. 85–101, Oct. 2024, doi: 10.4038/icter.v17i2.7279.

[47] M. Kamyab, G. Liu, and M. Adjeisah, "Attention-Based CNN and Bi-LSTM Model Based on TF-IDF and GloVe Word Embedding for Sentiment Analysis," Applied Sciences, vol. 11, no. 23, p. 11255, Nov. 2021, doi: 10.3390/app112311255.

[48] A. Vaswani et al., "Attention Is All You Need," arXiv (Cornell University), 2023.

[49] S. Islam et al., "A Comprehensive Survey on Applications of Transformers for Deep Learning Tasks," arXiv (Cornell University), 2023.

[50] A. Gillioz, J. Casas, E. Mugellini, and O. A. Khaled, "Overview of the Transformer-based Models for NLP Tasks," in Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, Sep. 2020, pp. 179–183. doi: 10.15439/2020F20.

[51] P. Cheng and K. Erk, "Attending to Entities for Better Text Understanding," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, pp. 7554–7561, Apr. 2020, doi: 10.1609/aaai.v34i05.6254.

[52] P. He, X. Liu, J. Gao, and W. Chen, "DeBERTa: Decoding-enhanced BERT with Disentangled Attention," arXiv (Cornell University), 2020.

[53] O. Rohanian et al., "Lightweight transformers for clinical natural language processing," Nat Lang Eng, vol. 30, no. 5, pp. 887–914, Sep. 2024, doi: 10.1017/S1351324923000542.

[54] Z. Gao, A. Feng, X. Song, and X. Wu, "Target-Dependent Sentiment Classification With BERT," IEEE Access, vol. 7, pp. 154290–154299, 2019, doi: 10.1109/ACCESS.2019.2946594.

[55] C. Sun, X. Qiu, Y. Xu, and X. Huang, "https://doi.org/10.48550/arxiv.1905.05583," arXiv (Cornell University), 2019.

[56] A. Marijić and M. Bagić Babac, "Predicting song genre with deep learning," Global Knowledge, Memory and Communication, vol. 74, no. 1/2, pp. 93–110, Jan. 2025, doi: 10.1108/GKMC-08-2022-0187.

[57] P. Delobelle, T. Winters, and B. Berendt, "RobBERT: a Dutch RoBERTa-based Language Model," in Findings of the Association for Computational Linguistics: EMNLP 2020, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 3255–3265. doi: 10.18653/v1/2020.findings-emnlp.292.

[58] H. Rehana et al., "Evaluating GPT and BERT models for protein–protein interaction identification in biomedical text," Bioinformatics Advances, vol. 4, no. 1, Jan. 2024, doi: 10.1093/bioadv/vbae133.

[59] H. Sohn and H. Lee, "MC-BERT4HATE: Hate Speech Detection using Multi-channel BERT for Different Languages and Translations," in 2019 International Conference on Data Mining Workshops (ICDMW), IEEE,

Nov. 2019, pp. 551–559. doi: 10.1109/ICDMW.2019.00084.

[60] X. Zheng, C. Zhang, and P. C. Woodland, “Adapting GPT, GPT-2 and BERT Language Models for Speech Recognition,” in 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), IEEE, Dec. 2021, pp. 162–168. doi: 10.1109/ASRU51503.2021.9688232.

[61] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” arXiv (Cornell University), 2013.

[62] J. Pennington, R. Socher, and C. Manning, “Glove: Global Vectors for Word Representation,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Stroudsburg, PA, USA: Association for Computational Linguistics, 2014, pp. 1532–1543. doi: 10.3115/v1/D14-1162.

[63] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Comput, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[64] K. Cho et al., “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Stroudsburg, PA, USA: Association for Computational Linguistics, 2014, pp. 1724–1734. doi: 10.3115/v1/D14-1179.

[65] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proceedings of the 2019 Conference of the North, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.

[66] Tom Brown et al., “Language Models are Few-Shot Learners,” Neural Information Processing Systems, vol. 33, pp. 1877–1901, 2020.

[67] C. Raffel et al., “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer ,” Journal of Machine Learning Research, vol. 21, 2020.

[68] OpenAI et al., “GPT-4 Technical Report,” arXiv (Cornell University), 2023.

[69] A. Y. Muaad et al., “Arabic Document Classification: Performance Investigation of Preprocessing and Representation Techniques,” Math Probl Eng, vol. 2022, pp. 1–16, Apr. 2022, doi: 10.1155/2022/3720358.

[70] K. M. Karaoglan and O. Findik, “Enhancing Aspect Category Detection Through Hybridised Contextualised Neural Language Models: A Case Study In Multi-Label Text Classification,” Comput J, vol. 67, no. 6, pp. 2257–2269, Jun. 2024, doi: 10.1093/comjnl/bxae004.

[71] C. Zeng, S. Li, Q. Li, J. Hu, and J. Hu, “A Survey on Machine Reading Comprehension—Tasks, Evaluation Metrics and Benchmark Datasets,” Applied Sciences, vol. 10, no. 21, p. 7640, Oct. 2020, doi: 10.3390/app10217640.

[72] B. Behera, G. Kumaravelan, and P. Kumar B., “Performance Evaluation of Deep Learning Algorithms in Biomedical Document Classification,” in 2019 11th International Conference on Advanced Computing (ICoAC), IEEE, Dec. 2019, pp. 220–224. doi: 10.1109/ICoAC48765.2019.246843.

[73] “IMDB Dataset of 50K Movie Reviews,” www.kaggle.com. https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

[74] “Sentiment Analysis on the Yelp Reviews Dataset,” kaggle.com. https://www.kaggle.com/code/omkarsabnis/sentiment-analysis-on-the-yelp-reviews-dataset

[75] “Amazon reviews,” www.kaggle.com. https://www.kaggle.com/datasets/kritanjalijain/amazon-reviews

[76] A. A. Jha, “Stanford Sentiment Treebank v2 (SST2),” Kaggle.com, 2020. https://www.kaggle.com/datasets/atulanandjha/stanford-sentiment-treebank-v2-sst2

[77] SetFit, “sst5,” Huggingface.co, Sep. 18, 2023. https://huggingface.co/datasets/SetFit/sst5

[78] A. Anand, “AG News Classification Dataset,” Kaggle.com, 2020. https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset/data

[79] “20 Newsgroups,” www.kaggle.com. https://www.kaggle.com/datasets/crawford/20-newsgroups

[80] The Devastator, “Reuters-21578 (Text Categorization),” Kaggle.com, 2022. https://www.kaggle.com/datasets/thedevastator/uncovering-financial-insights-with-the-reuters-2

[81] nasruddinaz, “BBCategorize - BBC News Category Classification,” Kaggle.com, Jul. 13, 2023. https://www.kaggle.com/code/nasruddinaz/bbcategorize-bbc-news-category-classification

[82] sbhatti, “News Summarization,” Kaggle.com, 2018. https://www.kaggle.com/datasets/sbhatti/news-summarization

[83] “Fake News Detection Datasets,” www.kaggle.com. https://www.kaggle.com/datasets/emineyetm/fake-news-detection-datasets

[84] “Fake News Challenge,” Fakenewschallenge.org, 2016. http://www.fakenewschallenge.org/

[85] Möbius, “COVID-19 Fake News Dataset,” Kaggle.com, 2020. https://www.kaggle.com/datasets/arashnic/covid19-fake-news

[86] Narendra Prasath, “MultiLabel Classification - Reuters News Dataset,” Kaggle.com, 2020. https://www.kaggle.com/datasets/narendrageek/reuters21578-multilabel-classification-news

[87] eshwarkoka, “GitHub - eshwarkoka/Medical-document-classification: Text classification on the medical abstracts in OHSUMED dataset,” GitHub, 2025. https://github.com/eshwarkoka/Medical-document-classification

[88] S. University, “Stanford Natural Language Inference Corpus,” Kaggle.com, 2015. https://www.kaggle.com/datasets/stanfordu/stanford-natural-language-inference-corpus

[89] “Welcome To Zscaler Directory Authentication,” Kaggle.com, 2026. https://www.kaggle.com/datasets/thedevastator/unlocking-language-understanding-with-the-multin

[90] “Question Pairs Dataset,” www.kaggle.com. https://www.kaggle.com/datasets/quora/question-pairs-dataset

[91] “The Stanford Question Answering Dataset,” rajpurkar.github.io. https://rajpurkar.github.io/SQuAD-explorer/

[92] “google/boolq · Datasets at Hugging Face,” huggingface.co, Jan. 02, 2024. https://huggingface.co/datasets/google/boolq

[93] facebookresearch, “GitHub - facebookresearch/XNLI: Evaluating Cross-lingual Sentence Representations,” GitHub, 2025. https://github.com/facebookresearch/XNLI

[94] facebookresearch, “GitHub - facebookresearch/MLDoc: A Corpus for Multilingual Document Classification in Eight Languages.,” GitHub, 2025. https://github.com/facebookresearch/MLDoc

[95] Y. Yang, Y. Zhang, C. Tar, and J. Baldridge, “PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification,” ACLWeb, Nov. 01, 2019. https://aclanthology.org/D19-1382/

[96] Z. Ji, Q. Wei, and H. Xu, “BERT-based Ranking for Biomedical Entity Normalization,” AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science, vol. 2020, pp. 269–277, 2020, Available: https://pubmed.ncbi.nlm.nih.gov/32477646/

[97] A. Mohan. kumar, “Legal Text Classification Dataset,” Kaggle.com, 2023. https://www.kaggle.com/datasets/amohankumar/legal-text-classification-dataset

[98] Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos, “MultiEURLEX - A multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer,” Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Jan. 2021, doi: https://doi.org/10.18653/v1/2021.emnlp-main.559.

[99] J. Yao and B. Yuan, “Optimization Strategies for Deep Learning Models in Natural Language Processing,” Journal of Theory and Practice of Engineering Science, vol. 4, no. 05, pp. 80–87, May 2024, doi: 10.53469/jtpes.2024.04(05).11.

[100] S. Huang, Y. Liu, C. Fung, H. Wang, H. Yang, and Z. Luan, “Improving Log-Based Anomaly Detection by Pre-Training Hierarchical Transformers,” IEEE Transactions on Computers, vol. 72, no. 9, pp. 2656–2667, Sep. 2023, doi: 10.1109/TC.2023.3257518.

[101] H. Ali et al., “Con-Detect: Detecting adversarially perturbed natural language inputs to deep classifiers through holistic analysis,” Comput Secur, vol. 132, p. 103367, Sep. 2023, doi: 10.1016/j.cose.2023.103367.

[102] Warto et al., “Systematic Literature Review on Named Entity Recognition: Approach, Method, and Application,” Statistics, Optimization & Information Computing, vol. 12, no. 4, pp. 907–942, Feb. 2024, doi: 10.19139/soic-2310-5070-1631.