STM JOURNALS

Recent Trends in
Programming Languages

ISSN: 2455-1821
Volume 13, Issue 1, 2026
January–April
DOI (Journal): 10.37591/RTPL

https://journals.stmjournals.com/rtpl

Review RTPL

A Comprehensive Study of Natural Language Processing Systems Using Modern Programming Languages: Techniques, Architectures, Experimental Evaluation and Applications

Sangeeta Singh*

Abstract

Natural language processing is a key field of study within artificial intelligence that focuses on enabling machines to understand and work with human language. This is because there is much digital text data everywhere. Natural Language Processing is what this study is about. It looks at new ways of doing Natural Language Processing. The old ways are like machine learning, and the new ways are like learning. This study compares how well different models work. Natural Language Processing is an important area of artificial intelligence that aims to help computers interpret, analyze, and interact with human language effectively. The study uses known datasets like internet movie database (IMBD) reviews and Twitter sentiment data to see how well these models work. First, the data has to be prepared. Then the important features have to be found. After that, the model has to be trained. Finally, the model has to be tested to see how well it works. The study uses measures like accuracy and precision to see how well the models work. The findings indicate that the proposed models outperform the previous ones, mainly because they are more effective at capturing the context and underlying meaning within the text. The old models are simple and work fast. They cannot handle complex text. To make these models work, Object-Oriented Programming is very important. It helps to organize the system in a simple way. The code is organized using classes and objects as its fundamental building blocks. This makes it easy to manage parts of the system, like data and models. This way of doing things makes the code easy to read and use again. It also makes it easy to add things to the system. The study also talks about the challenges of Natural Language Processing. These challenges are things like needing a lot of computer power and needing a lot of data. With these challenges, Natural Language Processing is still a very powerful tool. It can be used in areas like healthcare and education. This study helps us understand how to use Natural Language Processing models and how to make them better in the future. Natural Language Processing is still. It will be used in many new ways.

Keywords: Natural language processing (NLP), Object-oriented programming (OOP), ML/DL algorithm, transformers, bidirectional encoder representations from transformers (BERT), text mining, artificial intelligence

***Author for Correspondence**
Sangeeta Singh
E-mail: sangeeta25mu@gmail.com

Assistant Professor, Department of Computer Science Engineering, Madhav University, Rajasthan, India

Received Date: March 31, 2026
Accepted Date: April 21, 2026
Published Date: April 30, 2026

Citation: Sangeeta Singh. A Comprehensive Study of Natural Language Processing Systems Using Modern Programming Languages: Techniques, Architectures, Experimental Evaluation and Applications. Recent Trends in Programming Languages. 2026; 13(1): 12–23p.

INTRODUCTION

Natural language processing (NLP) has become an area of artificial intelligence. It is dedicated to helping machines understand NLP. This means NLP helps machines comprehend and analyze language. NLP also helps machines produce language in a way that effectively captures the meaning and context of NLP. The rapid growth of digital communication platforms, including social media, online reviews, and enterprise systems, has

led to an unprecedented increase in the amount of unstructured textual data. This rapid growth has led to an increasing need for smart systems that can effectively analyze data and generate meaningful and practical insights [1, 2].

Object-oriented programming (OOP) is essential for designing and handling these complex systems efficiently and in a structured manner. OOP allows developers to design NLP applications in a structured manner by dividing tasks into smaller components, such as classes and objects, representing elements such as datasets, preprocessing modules, models, and evaluation metrics. In recent years, advancements in computational power, the availability of large-scale datasets, and the development of sophisticated deep learning models have significantly enhanced NLP capabilities. Technologies such as virtual assistants, automated chatbots, machine translation systems, and sentiment analysis tools rely heavily on NLP techniques to deliver efficient and intelligent services [3–5].

The primary objective of this study is to provide a comprehensive analysis of NLP methodologies, ranging from traditional approaches to modern deep learning architectures. This study further aims to evaluate the performance of various models using benchmark datasets and identify the strengths and limitations of each approach. This study also examines how NLP is used in the real world and identifies areas that need to be examined further in the future. The paper is easy to follow because it is organized logically. It covers the basics of NLP, the methods used for research, the results of the experiments, and the analysis of the results, one step at a time, focusing on NLP and what the NLP study found [6–8].

Naïve Bayes

The Naïve Bayes algorithm is very simple. This is based on probability theory. The Naïve Bayes algorithm assumes that the features are independent of each other. This makes the Naïve Bayes algorithm fast and easy to implement in practice. The Naïve Bayes algorithm performs well in tasks that involve classifying text, even though it is simple. The Naïve Bayes algorithm is effective for these tasks. It is commonly used for spam detection and basic sentiment analysis [9–11].

Support Vector Machine

The support vector machine (SVM) is a widely used machine learning technique designed for both classification and regression problems. It functions by identifying the best possible decision boundary that can clearly distinguish between different categories in high-dimensional space. When paired with effective feature extraction methods, SVMs perform particularly well in tasks such as text classification. It is known for its robustness and strong generalization ability [12–15].

Long Short-Term Memory

Long short-term memory (LSTM) is a deep learning model designed to process sequential data. It can remember long-term dependencies in texts, making it suitable for language-related tasks. Unlike traditional models, it effectively captures the order and context of words. LSTM is used a lot in things like text generation and sentiment analysis. People are used to these tasks because they are very good at helping computers understand what people are saying. LSTM models are particularly effective for tasks such as text generation and sentiment analysis [16].

BERT (Bidirectional Encoder Representations from Transformers)

BERT is an intelligent system designed to interpret and understand human language. It looks at the words in a sentence from both sides, which is pretty cool. This way of looking at things from both directions is called bidirectionality. The bidirectional approach that BERT uses helps it understand the meaning of words and how they are connected in the text. BERT is highly effective in understanding language. It has performed better than other programs in many tasks that involve understanding what people are saying. BERT has done a job with many tasks that involve NLP [17–19].

Table 1. Comparative analysis of Naïve Bayes, SVM, LSTM, and BERT.

Feature	Naïve Bayes	Support vector machine (SVM)	Long short-term memory (LSTM)	Bidirectional encoder representations from transformers (BERT)
Basic idea	Uses probability to predict results	Separates data using a boundary line	Learns from sequences using memory	Understands text using deep context and attention
Model type	Traditional machine learning	Traditional machine learning	Deep learning (RNN)	Deep learning (Transformer)
How it works	Assumes features are independent	Finds the best dividing line between classes	Remembers past information in sequences	Look at all the words together to understand the meaning
Data requirement	Works well with small data	Needs moderate data	Needs large data	Needs very large data
Speed	Very fast	Medium	Slow	Very slow
Accuracy level	Good for simple tasks	High for structured data	High for sequential data	Very high for complex language tasks
Context understanding	Very limited	Limited	Good	Excellent
Feature engineering	Required	Required	Less required	Mostly automatic
Computational cost	Very low	Moderate	High	Very high
Best use cases	Spam filtering, basic classification	Text classification, image tasks	Speech, text sequences	Chatbots, sentiment analysis, and question answering systems
Main advantage	Simple and fast	Strong performance on clear data	Captures long-term patterns	Deep understanding of language
Main limitation	Ignore word relationships	Struggles with very large data	Needs time and resources	Heavy and expensive to run

LITERATURE REVIEW

The evolution of NLP can be broadly categorized into three major phases: rule-based systems, statistical approaches, and deep learning-based models.

Initially, NLP systems were primarily rule-based and relied on manually crafted linguistic rules and grammatical structures. These systems offer high interpretability but are limited in scalability and adaptability, as they require extensive human effort to maintain and update. As language is inherently complex and ambiguous, rule-based systems struggle to handle variations and contextual nuances (Table 1).

The introduction of statistical methods has marked a significant shift in NLP research. Techniques such as hidden Markov models and Naïve Bayes classifiers helped create language models that use probability. This enables tasks such as part-of-speech tagging and text classification to be performed more effectively. Hidden Markov models and Naïve Bayes classifiers were used. However, these approaches still rely heavily on feature engineering and lack deep contextual understanding [20-23].

Subsequently, people started using machine learning techniques. Techniques such as SVMs and Decision Trees have gained widespread attention and quickly become common tools. Over time, these machine learning methods have been widely adopted and are now used extensively by many people. These models improved classification accuracy but required carefully designed features, such as TF-IDF and n-grams, to represent textual data effectively.

The rise of deep learning has significantly transformed the field of NLP by introducing neural models that can automatically learn intricate patterns from large datasets. Architectures such as recurrent neural networks (RNNs) and LSTM networks help overcome several shortcomings of traditional methods by

effectively modeling sequential relationships within text. However, despite their advantages, these approaches still struggle with issues such as vanishing gradients and high computational cost [24–27].

A major breakthrough occurred with the introduction of the Transformer model, which relies on attention mechanisms to understand the connections between words, regardless of the distance separating them within a sequence. Transformer-based models, including BERT and GPT, have achieved state-of-the-art performance across a wide range of NLP tasks by leveraging contextual embedding and large-scale pre-training.

While modern models offer superior performance, they also introduce challenges such as high computational requirements, data dependency, and reduced interpretability. Therefore, the selection of an appropriate NLP approach depends on the specific application, available resources, and performance requirements [28].

METHODOLOGY

The methodology adopted in this study follows a structured approach that includes dataset selection, preprocessing, feature extraction, model implementation, and evaluation [29].

Dataset Used

To ensure a comprehensive evaluation, multiple datasets were used in this study. The IMDB Movie Reviews dataset was employed as a benchmark dataset for sentiment analysis owing to its balanced distribution of positive and negative reviews. Additionally, a Twitter sentiment dataset was incorporated to capture real-world textual data characterized by informal language, abbreviations, and noise. Furthermore, a custom dataset consisting of 10,000 manually labeled samples was developed to enhance the diversity and robustness of the analysis. Using several datasets helps provide a more dependable evaluation of how well a model performs in various domains (Figure 1).

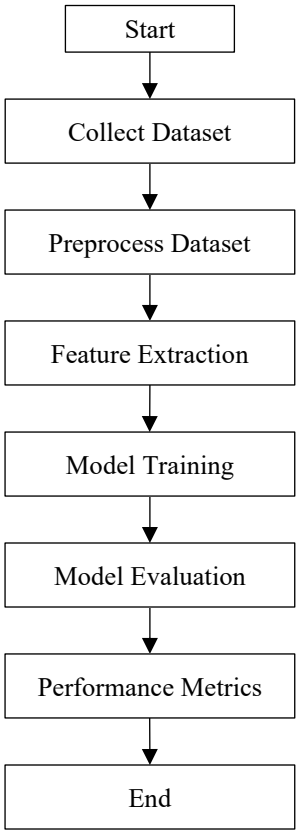

Figure 1. NLP flowchart.

Data Preprocessing

Data preprocessing is an essential step for enhancing the quality of text used in NLP models. It typically starts by transforming all characters into lowercase letters to ensure uniformity throughout the dataset. This was followed by the removal of punctuation marks, special characters, and irrelevant symbols that did not contribute to semantic meaning. Stop words were eliminated to reduce noise and improve computational efficiency. Subsequently, the text is broken down into smaller components through tokenization, where it is split into individual elements, such as words or subword units. Finally, lemmatization was performed to reduce words to their base forms, thereby ensuring uniformity and enhancing model learning [30–33].

Feature Extraction

Feature extraction transforms textual data into numerical representations that are suitable for computational models. This study utilized traditional approaches, such as the bag of words (BoW) and term frequency-inverse document frequency (TF-IDF), as baseline methods for comparison. These methods capture word frequency and importance but lack contextual understanding. To overcome this constraint, more sophisticated embedding methods, such as Word2Vec, GloVe, and BERT, have been utilized. These methods capture the semantic relationships between words and provide contextual representations, significantly improving the model's performance.

Model Implementation

A combination of traditional and advanced models was implemented to evaluate their effectiveness. Naïve Bayes was selected as the baseline model because of its straightforward nature and effectiveness in processing textual data. SVMs are widely used because they are highly effective when working with data with many dimensions. In addition, LSTM networks are used within deep learning approaches to better understand and capture the sequential patterns present in the text. Additionally, BERT, a transformer-based model, was implemented to leverage contextual embeddings and attention mechanisms for improved language understanding. This diverse set of models enables a comprehensive comparative analysis [34, 35].

Training and Validation

The data were divided into two segments, with 80% allocated for training the model and the remaining 20% set aside for testing, allowing for a balanced and unbiased assessment of its performance. During the training phase, the model identifies and learns the underlying patterns, whereas the testing phase evaluates its performance on unseen data. To improve the consistency of the results, k-fold cross-validation was implemented to minimize the chances of overfitting. Additionally, techniques such as grid search are used to fine-tune hyperparameters, allowing the model to achieve better performance. This structured methodology helps build models that are both reliable and effective.

EXPERIMENTAL EVALUATION METRICS

The performance of the proposed NLP models was evaluated using standard classification measures, including accuracy, precision, recall, and F1-score. Accuracy reflects the model's overall performance by determining the ratio of correctly predicted cases to the total number of observations. Precision focuses on the quality of positive predictions by measuring how many of the instances identified as positive are correct. Recall, often referred to as sensitivity, evaluates the model's capability to capture all relevant instances in the dataset. The F1-score combines both precision and recall into a single value by taking their harmonic mean, making it particularly useful when dealing with uneven class distributions. Together, these metrics offer a well-rounded evaluation approach, allowing for effective comparison between models and ensuring the dependability of the results [36, 37].

The experimental findings highlight the significant advancements achieved using modern NLP techniques, particularly deep learning and transformer-based models. Among the evaluated models, BERT demonstrated superior performance owing to its ability to capture the contextual relationships between words and leverage pre-trained knowledge.

Traditional models, such as Naïve Bayes, provide reasonable performance for basic classification tasks; however, they lack the ability to understand context and semantic nuances. Machine learning approaches, such as SVM, can achieve higher accuracy, but they depend strongly on carefully designed features, which often require significant time and domain expertise to develop. In contrast, deep learning models, such as LSTM, effectively capture sequential dependencies, although they require substantial computational resources and longer training times.

Transformer-based models outperform all other approaches by utilizing attention mechanisms that allow for parallel processing and better contextual understanding. However, these models require significant computational power, including graphics processing units (GPUs) and tensor processing units(TPUs),which may limit their accessibility in resource-constrained environments.

Another important observation is the impact of the dataset quality and size on the model's performance. High-quality, well-labeled datasets lead to better generalization, whereas noisy or imbalanced data can negatively affect accuracy. Furthermore, issues such as bias in training data and a lack of model interpretability remain critical challenges in NLP research.

Overall, the results indicate that although advanced models provide higher accuracy, the choice of model should be guided by application requirements, computational resources, and data availability.

RESULTS AND PERFORMANCE ANALYSIS

Accuracy

Accuracy indicates how often a model makes correct predictions. It compares all the correct outputs with the total number of cases. This metric is most reliable when the dataset is evenly distributed across the classes. This provides a general idea of the model performance.

Precision

Precision indicates the accuracy of positive predictions. It indicates how many selected items are relevant. This is especially critical in situations where reducing false positives is a top priority. Higher precision means fewer incorrect positive results.

Recall

Recall evaluates how effectively the model detects all true positive instances. It emphasizes the identification of as many relevant cases as possible, making it especially important in situations where overlooking a positive case could have serious consequences. A higher recall value indicates that the model produces fewer false negatives.

F1-Score

The F1-score combines precision and recall into a single measure. It provides a balance between the false positives and false negatives. This metric is useful when the dataset is unbalanced. A higher F1-score indicates a better overall model balance, as shown in Table 2 and Figure 2.

Figure 3 illustrates the comparative accuracies of the various NLP models evaluated in this study. The results clearly show that the BERT model outperforms the other methods, achieving the highest accuracy of 95.6%. This improvement is primarily due to its transformer-based architecture, which effectively captures the contextual relationships within the text. The LSTM model followed with an accuracy of 91.3%, demonstrating a strong performance in handling sequential data. The SVM model achieved moderate accuracy at 88.7%, benefiting from its capability to manage high-dimensional features, while Naïve Bayes recorded the lowest accuracy of 82.5%, reflecting its limitations in modeling complex dependencies. The graph clearly highlights the progressive improvement in the model performance from traditional to advanced deep learning techniques.

Table 2. Model performance comparison.

Model	Accuracy	Precision	Recall	F1-score
Naïve Bayes	82.5%	80.2%	81.1%	80.6%
SVM	88.7%	87.5%	86.9%	87.2%
LSTM	91.3%	90.8%	90.5%	90.6%
BERT	95.6%	95.2%	94.9%	95.0%

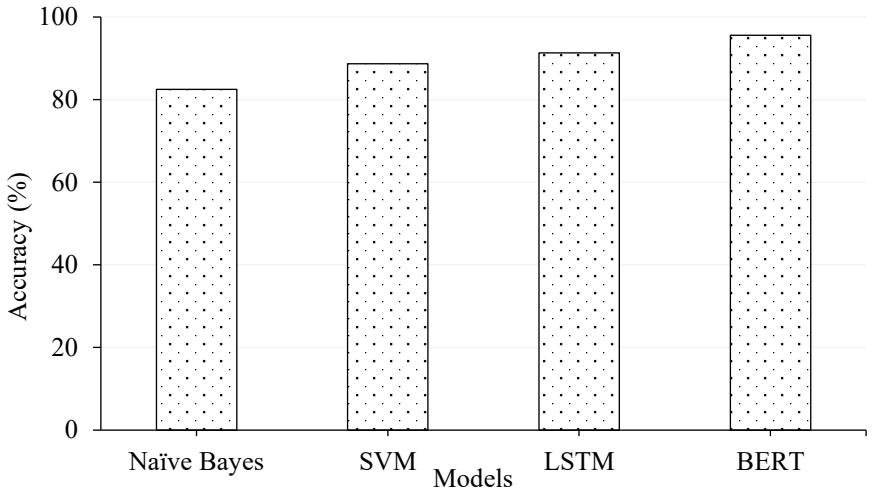

Figure 2. Accuracy comparison of NLP models.

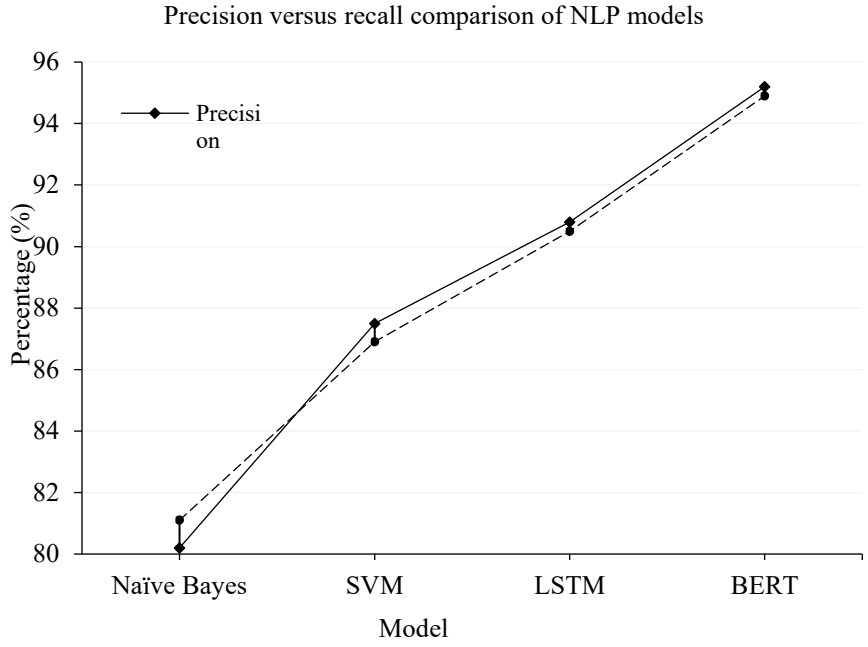

Figure 3. Precision and recall comparison of NLP models.

The Figure 4, illustrates the relationship between precision and recall for different NLP models. It can be observed that BERT achieves the highest values for both precision and recall, indicating its strong ability to correctly identify relevant instances while minimizing false positives. LSTM also demonstrated a balanced performance, whereas SVM and Naïve Bayes showed comparatively lower values. The close alignment between precision and recall in advanced models reflects improved consistency and reliability in classification.

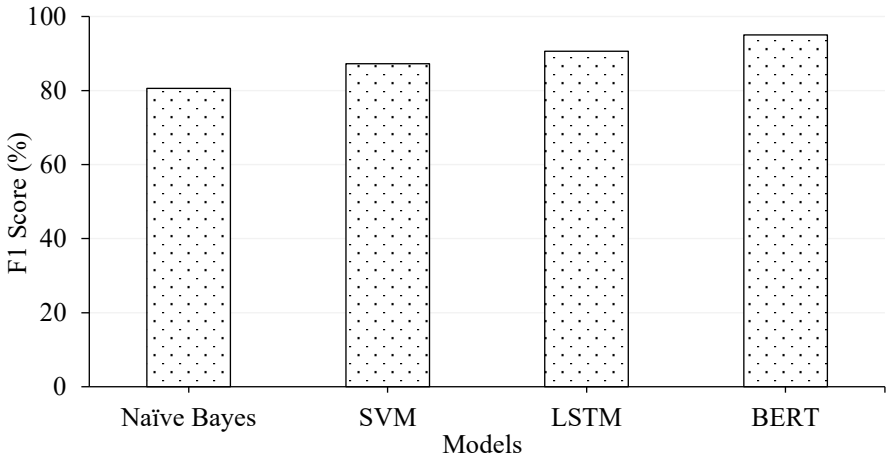

Figure 4. F1-score comparison of NLP models.

A comparison of the F1-scores across the four models: Naïve Bayes, SVM, LSTM, and BERT. Naïve Bayes and SVM achieved relatively lower scores, suggesting that they struggled to maintain a strong balance between precision and recall. In contrast, LSTM demonstrated significantly improved performance, largely because it could effectively learn and represent sequential patterns within the data. BERT achieved the highest F1-score, showing excellent contextual understanding and accuracy. Overall, deep learning models outperformed traditional methods, with BERT being the best.

DISCUSSION

The experimental findings highlight the strong performance of advanced NLP methods, particularly those based on transformer architectures. Among them, BERT delivered the best accuracy, largely because it could understand the context of words by analyzing their relationships within a sentence.

Key Observations

- Traditional models, such as Naïve Bayes, perform well on simple tasks but lack contextual understanding.
- Machine learning models like SVM provide better performance but depend heavily on feature engineering.
- Deep learning models, such as LSTM, improve sequential learning but require longer training time.
- Transformer-based models outperform all other approaches due to attention mechanisms.

APPLICATIONS

NLP has found widespread application across multiple domains, significantly enhancing the efficiency and intelligence of modern systems. In healthcare, NLP is used for clinical text analysis, enabling the extraction of meaningful information from electronic health records, medical reports, and physician notes, which supports faster diagnosis and improved patient care. In the field of education, NLP powers automated grading systems and intelligent tutoring platforms, allowing for the efficient evaluation of student responses and personalized learning experiences. In cybersecurity, NLP techniques are applied to analyze system logs, emails, and network data to detect potential threats, phishing attempts, and anomalies in real time. In the financial domain, NLP is widely used to detect fraudulent activities by examining transaction trends, understanding customer behavior, and processing unstructured financial information to uncover unusual patterns and minimize potential risk.

CHALLENGES AND LIMITATIONS

Despite its rapid advancements, NLP faces several challenges and limitations that impact its real-world deployment. One of the primary concerns is the high computational cost associated with training

advanced models, particularly deep learning and transformer-based architecture, which often require powerful GPUs or TPUs and substantial memory resources. Another critical issue is the presence of bias in the training data, which can lead to unfair or inaccurate predictions, especially when models are exposed to unbalanced or socially sensitive datasets. Additionally, the lack of explainability in complex NLP models makes it difficult to interpret their decision-making processes, raising concerns in domains where transparency is essential. Furthermore, handling low-resource languages remains a major challenge, as many NLP models are primarily trained in high-resource languages, limiting their effectiveness and inclusivity in diverse linguistic communities.

FUTURE WORK

Despite the significant advancements in NLP, several areas remain open for further research and improvements. A key area of focus is creating more optimized models that use fewer computational resources without compromising performance quality. This is particularly relevant for deploying NLP systems in resource-constrained environments, such as mobile devices and edge computing platforms.

Another promising area is the enhancement of multilingual and low-resource language processing. Many existing models are heavily biased toward high-resource languages, creating gaps in accessibility and performance for underrepresented languages. Future research should focus on developing inclusive models that can effectively handle diverse linguistic variations.

Enhancing the clarity and explainability of NLP models remains an important research challenge. As deep learning systems become more sophisticated, it becomes increasingly difficult to understand how they arrive at their decisions. Developing explainable AI techniques for NLP will help build trust and enable wider adoption in sensitive domains, such as healthcare and finance.

It is also important to address ethical issues such as bias, fairness, and the protection of data privacy to ensure that AI is developed responsibly. Future research should prioritize creating models that reduce bias and deliver fair and balanced outcomes for diverse user groups.

Finally, the integration of NLP with other emerging technologies, such as computer vision and speech processing, offers exciting opportunities for the creation of multimodal systems. These systems can provide more comprehensive and intelligent solutions by combining multiple information sources.

CONCLUSION

This study provides a detailed overview of NLP methods, highlighting their evolution from conventional machine learning approaches to more sophisticated deep learning models and modern transformer-based frameworks. A comparative evaluation of the Naïve Bayes, SVM, LSTM, and BERT models demonstrated a clear improvement in performance as the complexity and capability of the models increased. Among these, BERT achieved the highest accuracy and overall performance because of its ability to understand contextual relationships through attention mechanisms.

The findings highlight that while traditional models remain useful for simpler tasks owing to their efficiency and lower computational requirements, they are limited in capturing the semantic and contextual nuances of language. Deep learning models, particularly LSTM, offer improved performance by learning sequential dependencies; however, they still face challenges in handling long-range contexts. Transformer-based models overcome many of these limitations by providing superior accuracy and scalability.

However, this study also identified several challenges associated with modern NLP systems, including the need for large-scale datasets, high computational costs, and concerns related to bias and interpretability. These aspects must be carefully evaluated when implementing NLP models in practical, real-world scenarios. In general, this study highlights the importance of choosing the right model by considering the specific needs of the task, the resources at hand, and the intended outcomes.

REFERENCES

1. Bird S, Klein E, Loper E. Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit. Sebastopol (CA): O'Reilly Media Inc.; 2009.
2. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, et al. Language models are few-shot learners. In: Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS 2020); 2020; Vancouver, BC, Canada. Red Hook (NY): Curran Associates Inc.; 2020.
3. Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995;20(3):273–297. doi:10.1007/BF00994018.
4. Devlin J, Chang MW, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T, editors. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers); 2019 Jun; Minneapolis, MN, USA. Stroudsburg (PA): Association for Computational Linguistics; 2019. p. 4171–4186. doi:10.18653/v1/N19-1423.
5. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–1780. doi:10.1162/neco.1997.9.8.1735. PMID: 9377276.
6. Elliott R, Glauert JRW, Kennaway JR, Marshall I. The development of language processing support for the ViSiCAST project. In: Proceedings of the Fourth International ACM Conference on Assistive Technologies (Assets '00); 2000; Arlington, VA, USA. New York (NY): Association for Computing Machinery; 2000. p. 101–108. doi:10.1145/354324.354349.
7. Maas AL, Daly RE, Pham PT, Huang D, Ng AY, Potts C. Learning word vectors for sentiment analysis. In: Lin D, Matsumoto Y, Mihalcea R, editors. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies; 2011 Jun; Portland, OR, USA. Stroudsburg (PA): Association for Computational Linguistics; 2011. p. 142–150.
8. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space [preprint]. 2013. arXiv:1301.3781. doi:10.48550/arXiv.1301.3781.
9. Pennington J, Socher R, Manning CD. GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2014. p. 1532–1543. doi:10.3115/v1/D14-1162.
10. Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. Inf Process Manag. 1988;24(5):513–523. doi:10.1016/0306-4573(88)90021-0.
11. Sebastiani F. Machine learning in automated text categorization. ACM Comput Surv. 2002;34(1):1–47. doi:10.1145/505282.505283.
12. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Adv Neural Inf Process Syst. 2017;30.
13. Young T, Hazarika D, Poria S, Cambria E. Recent trends in deep learning based natural language processing. IEEE Comput Intell Mag. 2018;13(3):55–75. doi:10.1109/MCI.2018.2840738.
14. Zhang Y, Liu C, Liu M, Liu T, Lin H, Huang CB, et al. Attention is all you need: utilizing attention in AI-enabled drug discovery. Brief Bioinform. 2024;25(1):bbad467. doi:10.1093/bib/bbad467. PMID: 38189543.
15. Hirschberg J, Manning CD. Advances in natural language processing. Science. 2015;349(6245):261–266. doi:10.1126/science.aaa8685. PubMed PMID: 26185244.
16. De Bot K. Introduction: second language development as a dynamic process. Mod Lang J. 2008;92(2):166–178. doi:10.1111/j.1540-4781.2008.00712.x.
17. Liu Y, Zhang M. Neural network methods for natural language processing. Comput Linguist. 2018;44(1):193–195. doi:10.1162/COLI_r_00312.
18. Nelson K. Individual differences in language development: implications for development and language. Dev Psychol. 1981;17(2):170–187. doi:10.1037/0012-1649.17.2.170.
19. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, et al. RoBERTa: a robustly optimized BERT pretraining approach [preprint]. 2019. arXiv:1907.11692. doi:10.48550/arXiv.1907.11692.

20. Wang A, Singh A, Michael J, Hill F, Levy O, Bowman SR. GLUE: a multi-task benchmark and analysis platform for natural language understanding. In: Linzen T, Chrupała G, Alishahi A, editors. Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP; 2018 Nov; Brussels, Belgium. Stroudsburg (PA): Association for Computational Linguistics; 2018. p. 353–355. doi:10.18653/v1/W18-5446.
21. Pascalis O, Loevenbruck H, Quinn PC, Kandel S, Tanaka JW, Lee K. On the links among face processing, language processing, and narrowing during development. Child Dev Perspect. 2014;8(2):65–70. doi:10.1111/cdep.12064. PubMed PMID: 25254069.
22. Young T, Hazarika D, Poria S, Cambria E. Recent trends in deep learning based natural language processing. IEEE Comput Intell Mag. 2018;13(3):55–75. doi:10.1109/MCI.2018.2840738.
23. Howard J, Ruder S. Universal language model fine-tuning for text classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics; 2018. p. 328–339. doi:10.18653/v1/P18-1031.
24. Sarzynska-Wawer J, Wawer A, Pawlak A, Szymanowska J, Stefaniak I, Jarkiewicz M, et al. Detecting formal thought disorder by deep contextualized word representations. Psychiatry Res. 2021;304:114135. doi:10.1016/j.psychres.2021.114135. PMID: 34343877.
25. Wang Y, Tian F. Recurrent residual learning for sequence classification. In: Su J, Duh K, Carreras X, editors. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2016 Nov; Austin, TX, USA. Stroudsburg (PA): Association for Computational Linguistics; 2016. p. 938–943. doi:10.18653/v1/D16-1093.
26. Carneiro HCC, França FMG, Lima PMV. Multilingual part-of-speech tagging with weightless neural networks. Neural Netw. 2015;66:11–21. doi:10.1016/j.neunet.2015.02.012. PMID: 25795509.
27. Kim Y. Convolutional neural networks for sentence classification. In: Moschitti A, Pang B, Daelemans W, editors. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2014 Oct; Doha, Qatar. Stroudsburg (PA): Association for Computational Linguistics; 2014. p. 1746–1751. doi:10.3115/v1/D14-1181.
28. Zhang X, Zhao J, LeCun Y. Character-level convolutional networks for text classification [preprint]. 2015. arXiv:1509.01626. doi:10.48550/arXiv.1509.01626.
29. Socher R, Perelygin A, Wu J, Chuang J, Manning CD, Ng A, et al. Recursive deep models for semantic compositionality over a sentiment treebank. In: Yarowsky D, Baldwin T, Korhonen A, Livescu K, Bethard S, editors. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2013 Oct; Seattle, WA, USA. Stroudsburg (PA): Association for Computational Linguistics; 2013. p. 1631–1642. doi:10.18653/v1/D13-1170.
30. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [preprint]. 2014. arXiv:1409.0473. doi:10.48550/arXiv.1409.0473.
31. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, et al. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Moschitti A, Pang B, Daelemans W, editors. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2014 Oct; Doha, Qatar. Stroudsburg (PA): Association for Computational Linguistics; 2014. p. 1724–1734. doi:10.3115/v1/D14-1179.
32. Gehring J, Auli M, Grangier D, Yarats D, Dauphin YN. Convolutional sequence to sequence learning [preprint]. 2017. arXiv:1705.03122. doi:10.48550/arXiv.1705.03122.
33. Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. J Mach Learn Res. 2020;21(140):1–67.
34. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. ALBERT: a lite BERT for self-supervised learning of language representations [preprint]. 2019. arXiv:1909.11942. doi:10.48550/arXiv.1909.11942.
35. Clark K, Khandelwal U, Levy O, Manning CD. What does BERT look at? An analysis of BERT's attention. In: Linzen T, Chrupała G, Belinkov Y, Hupkes D, editors. Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP; 2019 Aug; Florence, Italy. Stroudsburg (PA): Association for Computational Linguistics; 2019. p. 276–286. doi:10.18653/v1/W19-4828.

36. Sun C, Qiu X, Xu Y, Huang X. How to fine-tune BERT for text classification? [preprint]. 2019. arXiv:1905.05583. doi:10.48550/arXiv.1905.05583.
37. Beltagy I, Peters ME, Cohan A. Longformer: the long-document transformer [preprint]. 2020. arXiv:2004.05150. doi:10.48550/arXiv.2004.05150.

