International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

Deep Learning Techniques for Personality Prediction: A Comparative Study on Architectural Complexity and Performance

Yasmeen Banu M
Research Scholar
Dept. of Computer Science & Engg.
Hindustan Institute of Tech. & Science
Chennai, Tamil Nadu, India
Orcid ID : 0009-0000-3626-2334

Dr. T. Sudalaimuthu
Professor (Supervisor / Guide)
Dept. of Computer Science & Engg.
Hindustan Institute of Tech. & Science
Chennai, Tamil Nadu, India
Orcid ID : 0000-0003-0371-9371

Dr. N. Muthuvairavan Pillai
Assoc. Prof. (Research Supervisor)
Dept. of Computer Science & Engg.
R.M.D. Engineering College
Chennai, Tamil Nadu, India

Abstract: Personality prediction aims to infer an individual‟s psychological characteristics from behavioural and textual information. It has become an important research area with potential applications in recruitment, education, psychological assessment, marketing, and human–computer interaction. Conventional machine learning methods commonly depend on manually engineered representations such as Bag-of-Words, TF-IDF, and Linguistic Inquiry and Word Count (LIWC). Although these representations can be effective, they provide limited capability for modelling word order, long-range dependencies, and contextual meaning. Deep learning methods overcome several of these limitations by learning task-relevant representations directly from the input data. This paper presents a comparative study of major deep learning architectures applied to text-based personality prediction. Six representative approaches are considered: Convolutional Neural Networks (CNN), Long Short-Term Memory networks (LSTM), Bidirectional LSTM (BiLSTM), Gated Recurrent Units (GRU), hybrid CNN-RNN architectures, and Transformer-based models. The comparison examines not only predictive performance but also architectural complexity, computational requirements, data requirements, robustness, and interpretability. A unified framework is developed to organize the complete prediction pipeline, including data acquisition, preprocessing, text representation, model training, validation, and evaluation. The study considers personality prediction under both the Myers-Briggs Type Indicator (MBTI) and Big Five personality frameworks and reviews the use of static representations such as Word2Vec and GloVe alongside contextual representations produced by BERT and RoBERTa. The comparative analysis indicates that contextual Transformer representations generally provide stronger predictive performance than models relying solely on randomly initialized or static embeddings. Fully fine-tuned Transformer models, particularly BERT- and RoBERTa-based approaches, demonstrate strong performance when sufficient computational resources and labelled data are available. Hybrid approaches and BiLSTM models provide a practical compromise between contextual modelling capability and computational cost, whereas CNN-based models remain attractive for resource-constrained environments because of their relatively low complexity and faster training. The findings demonstrate that architecture selection should not depend on accuracy alone. Instead, computational resources, dataset size, retraining requirements, inference cost, and interpretability should also be considered when selecting a personality-prediction model for practical deployment.

Keywords : Personality Prediction, Deep Learning, Convolutional Neural Network, Recurrent Neural Network, Long Short-Term Memory, Bidirectional LSTM, Gated Recurrent Unit, CNN-RNN, Transformer, BERT, RoBERTa, Word Embeddings, Attention Mechanism, Text Classification

I. INTRODUCTION

Personality represents a set of relatively consistent characteristics that influence how individuals think, communicate, respond to situations, and interact with others. Reliable information about personality can be useful across several domains. In recruitment, it may support the assessment of job-related characteristics; in education, it can contribute to personalised learning strategies; in psychological and clinical settings, it may provide additional information for assessment; and in marketing and human–computer interaction, it can support more personalised interactions and recommendations.Traditional personality assessment has largely relied on structured questionnaires and self-reported responses. Two widely studied frameworks are the Big Five model and the Myers-Briggs Type Indicator (MBTI). The Big Five framework describes personality

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

through Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism (OCEAN), whereas MBTI represents personality through four preference dimensions: Introversion/Extraversion,Sensing/Intuition,Thinking/Feelin g, and Judging/Perceiving. Although questionnaire-based assessment is well established, it depends on participants providing responses that accurately represent their behaviour and preferences. Large-scale administration can also be time-consuming, while responses may be influenced by social desirability or by the way individuals perceive and present themselves.The expansion of digital communication has created an alternative source of information for personality analysis. Written material such as essays, emails, online discussions, and social-media posts contains patterns of vocabulary, syntax, expression, and communication style that may provide information relevant to personality prediction. This has encouraged researchers to investigate automated approaches capable of inferring personality characteristics from naturally generated text rather than relying exclusively on explicit questionnaires.Early automated systems generally used conventional machine learning techniques together with manually constructed textual features. Representations such as Bag-of-Words, TF-IDF, and Linguistic Inquiry and Word Count (LIWC) provide useful numerical descriptions of language, but they have limitations when complex relationships within a document need to be captured. In particular, these representations provide limited information about word order, dependencies between distant words, and the contextual meaning of expressions. Consequently, conventional feature-based approaches may not adequately represent the richer linguistic patterns associated with personality-related behaviour. Deep learning offers a different approach by allowing useful representations to be learned directly from the input data. CNN-based models can identify local combinations of words and other short-range patterns within text representations. Recurrent architectures, including LSTM, BiLSTM, and GRU, are designed to model sequential information and can retain information from earlier portions of a text while processing later content. Transformer architectures extend contextual modelling through self-attention, allowing relationships between different parts of a sequence to be represented without relying exclusively on sequential recurrence. Pre-trained language models such as BERT can further exploit representations learned from large-scale unlabelled text and adapt them to personality-classification tasks through fine-tuning.The increasing capability of deep learning models, however, introduces practical challenges. Compared with simpler machine learning approaches, deep neural networks may require greater computational resources, larger training datasets, and longer training periods. Selecting appropriate architectures and hyperparameters can also increase development cost. Furthermore, complex neural models are generally more difficult to interpret, which is particularly important when personality prediction is considered for applications such as recruitment or psychological assessment, where model decisions may affect individuals.

Existing research has investigated a wide range of CNN, recurrent, hybrid, and Transformer-based architectures for personality prediction. However, reported performance differs substantially across studies because experiments often use different datasets, personality frameworks, representations, preprocessing procedures, and training configurations. Therefore, an accuracy value reported for one architecture cannot necessarily be interpreted as a direct indication of its superiority over another architecture evaluated under different conditions.For practical deployment, predictive accuracy is consequently only one part of the decision. Other factors, including computational requirements, training and inference time, amount of labelled data, robustness, model complexity, and interpretability, can determine whether an architecture is appropriate for a particular application. A model that achieves high accuracy under a resource-rich experimental environment may not necessarily be the most suitable option for an application requiring frequent retraining or operation with limited computational resources.This paper addresses these considerations through a structured comparative analysis of six major deep learning approaches for personality prediction: CNN, LSTM, BiLSTM, GRU, hybrid CNN-RNN architectures, and Transformer-based models. The study examines their reported predictive performance together with computational requirements, data requirements, robustness, and interpretability. A unified framework is used to organize the stages of personality prediction from data acquisition and preprocessing through representation learning, model development, training, validation, and evaluation. The objective is to determine how architectural choice influences both predictive capability and practical deployment suitability.The remainder of this paper is organized to progressively develop this comparison. Section II reviews existing research on deep learning for personality prediction. Section III examines the major architecture families in greater detail. Section IV compares deep learning approaches with traditional machine learning methods. Sections V and VI define the research problem and identify the major research gaps. Section VII introduces the proposed comparative framework, while Sections VIII and IX describe its operational workflow and evaluation procedure. Section X presents the mathematical formulations of the candidate architectures, and Section XI describes the datasets and experimental considerations. Sections XII–XIV discuss comparative performance, computational requirements, and interpretability. Section XV presents the architecture-selection recommendations, followed by the embedding ablation analysis in Section XVI and the threats to validity in Section XVII. Section XVIII concludes the paper and identifies directions for future work, while Section XIX provides practical deployment guidance.

CONTRIBUTIONS

The principal contributions of this study are summarized below:

1. **Structured review of deep learning architectures:**

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

The study provides an organized review of major neural architectures used for text-based personality prediction, covering CNN, LSTM, BiLSTM, GRU, hybrid CNN-RNN models, and Transformer-based approaches within the BERT family.

2. **Unified personality-prediction framework:** A common framework is developed to represent the complete prediction process, beginning with data acquisition and preprocessing and continuing through text representation, model training, validation, and performance evaluation.
3. **Comparison of datasets and representation strategies:**
   The study consolidates findings from research using datasets such as the MBTI Kaggle corpus, Essays/Big-Five dataset, PANDORA, myPersonality, and multimodal datasets. It also examines the distinction between static word representations, including Word2Vec and GloVe, and contextual representations generated by BERT and RoBERTa.
4. **Joint analysis of performance and model complexity:** Rather than treating accuracy as the sole criterion, the comparison considers parameters, training and inference requirements, computational cost, labelled-data requirements, robustness, and interpretability. This provides a broader basis for assessing the practical suitability of each architecture.
5. **Practical architecture recommendations:** Based on the combined evidence, the study identifies the lightweight architecture, the architecture offering the strongest predictive performance, and the approach providing a practical balance between accuracy, computational requirements, and deployment considerations.

II. LITERATURE REVIEW

Personality prediction from written language has been investigated through both conventional machine-learning methods and more recent deep-learning approaches. Earlier studies commonly represented text using manually constructed features such as Bag-of-Words, TF-IDF, and linguistic indicators. While these representations can capture useful statistical and lexical information, they have limitations in modelling the broader meaning of language and relationships between distant words. As a result, researchers increasingly shifted toward deep-learning techniques capable of learning relevant representations directly from textual data.

One of the early deep-learning approaches applied to personality prediction was the Convolutional Neural Network (CNN). In text-based applications, CNNs process sequences of word representations through convolutional filters that can detect informative local structures, including keywords, short expressions, and neighbouring word patterns. Such local linguistic features can contain useful signals associated with personality traits. Research involving essays and social-media text has demonstrated that CNN-based models can provide competitive classification performance and serve as efficient baselines for evaluating more complex architectures [18].

Because language is inherently sequential, researchers also explored Recurrent Neural Networks (RNNs) for personality prediction. Unlike CNNs, recurrent models process textual information in sequence and can incorporate information from earlier words while analysing later parts of the input. However, standard RNNs often struggle to preserve useful information across long sequences. To overcome this limitation, gated architectures such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Bidirectional LSTM (BiLSTM) were introduced. In one comparative study, LSTM achieved better results than RNN, GRU, and BiLSTM under the specific experimental conditions considered [2]. This indicates that increasing architectural complexity does not necessarily produce better personality-prediction performance.

Another important direction in this field involves hybrid neural architectures. CNN-RNN combinations attempt to exploit the advantages of both model families: convolutional layers identify local linguistic patterns, while recurrent layers model dependencies across the sequence. This combination can be useful for personality prediction because personality-related characteristics may appear in both individual phrases and the broader structure of a person's writing. Previous studies have investigated attention-based hierarchical models and CNN-BiLSTM architectures, reporting competitive results on social-media and other text-based datasets [3], [4].

The emergence of Transformer-based language models introduced a further development in personality prediction. Unlike static word representations, contextual models generate representations that depend on the surrounding linguistic context. Consequently, the same word may be represented differently depending on how it is used within a sentence. Models such as BERT can therefore capture complex relationships within text more effectively than many traditional embedding approaches. Researchers have combined BERT with linguistic and psycholinguistic features to predict both Big Five and MBTI personality characteristics [5]. Other studies have used BERT-generated representations together with CNN, LSTM, or GRU classification layers. These approaches aim to combine strong contextual language representations with relatively lightweight prediction architectures. In several studies, BERT-CNN models have shown strong performance for personality-classification tasks involving MBTI and Big Five traits [1], [6]. These findings suggest that improvements in text representation can sometimes contribute more substantially to performance than simply increasing the complexity of the classification network.

Dataset characteristics also play an important role in determining model performance. Personality datasets frequently contain uneven class distributions, where certain personality categories are represented by substantially more samples than others. This imbalance can influence

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

evaluation results and may favour models that perform well on majority classes. Researchers have therefore explored methods such as resampling and other class-balancing strategies. A comparative study involving MLP, LSTM, GRU, 1D-CNN, and LSTM-CNN models showed that performance could vary according to the technique used to address class imbalance [9]. Therefore, reported accuracy should always be interpreted together with information about the dataset, preprocessing procedure, and evaluation methodology.

III. EXTENDED RELATED WORK BY ARCHITECTURE FAMILY

A. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) have been applied to text-based personality prediction by learning informative patterns directly from numerical representations of text. In a typical text-classification setting, convolutional filters operate over sequences of word or token representations to identify discriminative local patterns. These patterns may correspond to individual words, short phrases, or combinations of neighbouring tokens that provide useful information for distinguishing personality characteristics.

One advantage of CNN-based models is their relatively efficient computation, since convolutional operations can be performed in parallel. This makes CNNs suitable for personality-prediction applications where computational resources or training time are constrained. Previous studies have demonstrated that CNN-based approaches can achieve useful classification performance on essay and social-media datasets [18]. However, CNNs primarily focus on local patterns and therefore have limitations when personality-related information depends on relationships between words that occur far apart in a document. Increasing the convolutional receptive field or stacking additional layers can partially address this limitation, but may also increase model complexity. Consequently, CNNs are generally more appropriate when local linguistic patterns provide sufficient information for the prediction task, while architectures with explicit sequence or contextual modelling may be preferable for longer-range dependencies.

B. Recurrent Neural Networks and Gated Variants

Recurrent Neural Networks (RNNs) provide an alternative approach by explicitly modelling the sequential structure of text. The input is processed progressively, allowing information from earlier tokens to influence the representation of later tokens. This sequential formulation is useful for personality prediction because linguistic characteristics can depend not only on individual words but also on their ordering and surrounding context. A major limitation of conventional RNNs is their difficulty in preserving information over long sequences. Long Short-Term Memory (LSTM) networks address this problem through gated memory mechanisms that regulate the information retained and passed through the network. Gated Recurrent Units (GRUs) follow a similar principle but employ a comparatively simpler gating structure, which can reduce the number of parameters and computational requirements. Bidirectional LSTM (BiLSTM) models extend this approach by processing the input sequence in both forward and backward directions. As a result, the representation of a token can incorporate information from both preceding and following parts of the text. This capability can be beneficial when personality-related linguistic cues depend on context occurring on either side of a particular word or phrase. Despite these advantages, recurrent architectures do not consistently provide the best performance across all personality-prediction datasets. Their effectiveness can depend on dataset size, text characteristics, embedding strategy, personality framework, and training configuration [2], [15]. Therefore, comparisons between LSTM, GRU, and BiLSTM models should consider not only predictive performance but also computational complexity and the characteristics of the target dataset.

C. Hybrid CNN-RNN Architectures

Hybrid CNN-RNN architectures combine complementary capabilities of convolutional and recurrent networks. In a commonly used configuration, the CNN component first extracts informative local features from the textual representation, after which the recurrent component models relationships among the extracted features. This arrangement allows the model to consider both short-range linguistic patterns and sequential dependencies within the input text.

The combination is particularly relevant to personality prediction because personality-related information may be expressed through specific words or phrases while also being reflected in the broader organization and progression of a person's writing. CNN layers can therefore provide compact local representations, whereas recurrent layers can capture dependencies between these representations across the sequence. Previous studies have explored CNN-BiLSTM and related hybrid architectures for personality recognition and have reported competitive results on social-media and other text-based datasets [3], [4]. However, the additional network components also increase the number of parameters and may result in greater training and inference costs. Thus, the benefit of combining CNN and recurrent processing should be evaluated against the additional computational requirements rather than assuming that increased architectural complexity will always produce better predictions.

D. Transformer-Based Architectures

Transformer-based models represent a further development in text-based personality prediction by using self-attention to model relationships between different parts of an input sequence. Unlike static word representations, contextual representations generated by Transformer models can vary according to the surrounding linguistic context. Consequently, the same word may contribute differently to the representation depending on how it is used within a

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

sentence or document. Models such as BERT and RoBERTa use large-scale pre-training to obtain contextual language representations that can subsequently be fine-tuned or used as feature extractors for personality-related classification tasks [5], [7], [8], [12]. Researchers have also combined Transformer representations with CNN, LSTM, and GRU components to obtain contextual features while maintaining comparatively lightweight downstream classification structures [1], [6]. Transformer-based approaches can provide strong predictive performance, particularly when sufficient labelled data and computational resources are available. However, their advantages are accompanied by higher computational requirements and increased model complexity compared with simpler CNN or recurrent approaches. Their internal decision-making can also be more difficult to interpret, which is an important consideration for personality prediction applications. Therefore, Transformer architectures should not be evaluated solely according to accuracy. Computational cost, dataset size, training requirements, inference efficiency, and interpretability should also be considered when determining their suitability for a particular personality-prediction scenario.

E. Comparative Perspective

The reviewed architecture families represent different approaches to learning personality-related information from text. CNNs emphasize local linguistic patterns and offer comparatively efficient computation, while RNN-based architectures focus more explicitly on sequential dependencies. Hybrid CNN-RNN models attempt to combine both characteristics, whereas Transformer-based models provide contextual representations through self-attention. Importantly, the performance difference between these architectures may also be influenced by the underlying text representation. Consequently, an architecture that performs well with one embedding strategy or dataset may not necessarily achieve the same advantage under another experimental configuration. This observation supports the need for controlled comparisons in which representation strategy, data preprocessing, class distribution, and evaluation methodology are considered alongside the neural architecture itself.

IV. DEEP LEARNING VERSUS TRADITIONAL MACHINE LEARNING

Comparing deep learning with traditional machine-learning approaches remains important because conventional methods are still widely used for personality prediction. Algorithms such as Naive Bayes, Logistic Regression, and Support Vector Machines commonly rely on manually prepared features, including TF-IDF representations and linguistic features such as LIWC. These approaches generally require fewer computational resources, are easier to interpret, and can remain effective when the amount of labelled training data is limited. Previous research has reported accuracy values ranging approximately from 60% to 82% for traditional machine-learning methods applied to MBTI and Essays datasets [20]. Deep-learning approaches offer a different strategy by learning useful representations directly from data. Their advantage becomes particularly evident when sufficiently informative representations and adequate training resources are available. However, deep models are also more sensitive to factors such as dataset size, model capacity, and training configuration. When only a small labelled dataset is available, highly parameterized models may overfit and fail to generalize effectively. CNN and recurrent architectures trained from scratch must learn both general linguistic patterns and personality-related characteristics from the available dataset. This can be challenging when the training corpus is limited. Pre-trained Transformer models address this problem differently. Before adaptation to personality prediction, they acquire general knowledge about language from large-scale text collections. During fine-tuning, this previously learned knowledge is adjusted to establish relationships between the language representations and personality labels. As a result, pre-trained Transformers may provide an advantage when labelled personality data is limited. This study considers six representative deep-learning architecture families: CNN, LSTM, BiLSTM, GRU, hybrid CNN-RNN models, and fine-tuned Transformer-based models. Together, these architectures represent a progression from relatively lightweight neural networks to advanced contextual language models. Evaluating them within a common framework makes it possible to examine not only predictive performance but also the effects of model complexity, computational requirements, and data availability.

V. PROBLEM STATEMENT

Research on personality prediction has produced a wide range of reported performance values. For example, accuracy may fall within the lower range for small or highly imbalanced MBTI datasets, while substantially higher results may be reported when larger datasets, improved representations, and more effective training procedures are used [2], [7]. Similar variation can be observed across CNN and hybrid CNN-RNN models, whose performance is influenced by embedding selection, dataset quality, preprocessing decisions, and class-distribution characteristics [1], [3], [4], [9]. For this reason, directly comparing accuracy values reported by separate studies can lead to misleading conclusions. A model that performs strongly on one dataset may produce substantially different results when evaluated on another dataset or under a different preprocessing and validation strategy. Another limitation in existing research is the strong emphasis on accuracy and F1-score. Although these measures are important, they do not completely describe whether a model is suitable for practical deployment. Model size, training duration, GPU requirements, inference speed, availability of labelled data, and interpretability can also influence the choice of architecture.

Based on these observations, this study addresses two central issues:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

- **To organize existing deep-learning research on personality prediction within a common comparative structure**, enabling different architectures and their reported outcomes to be examined more systematically.
- **To evaluate practical requirements together with predictive performance**, so that architecture selection considers computational resources, data availability, model complexity, and deployment requirements rather than accuracy alone.

Illustrative Scenario

Consider two organizations planning to develop personality-prediction systems using the same set of deep-learning architectures.

The first organization has access to multiple GPUs, a large labelled dataset, and sufficient time for extensive experimentation. Under these conditions, computationally demanding approaches such as a fine-tuned Transformer or a BERT-based architecture combined with an LSTM classifier may be practical if the additional computational cost produces improved predictive performance.

The second organization operates with fewer computational resources, a smaller labelled dataset, and a requirement for frequent model updates. In this situation, deploying a large Transformer may not be the most efficient option. A comparatively lightweight CNN or BiLSTM using pre-trained and frozen embeddings may provide a more practical compromise between predictive capability and computational cost. This demonstrates why the selection of a personality-prediction architecture should depend on the intended deployment environment rather than assuming that the most complex model is always the best choice.

VI. RESEARCH GAP

Although deep learning has shown promising results for automated personality prediction from text, several limitations remain in the existing research. Previous studies have examined CNN, LSTM, GRU, BiLSTM, hybrid CNN-RNN, and Transformer-based architectures; however, these approaches are often evaluated using different datasets, preprocessing procedures, representation strategies, and evaluation settings. Such differences make it difficult to determine whether variations in reported performance are primarily attributable to the architecture or to the experimental configuration.

A. Difficulty in Separating Representation Quality from Architecture Quality

A significant gap exists in distinguishing the contribution of the **text representation** from that of the **classification architecture**. In many existing studies, changes in embedding strategy and model architecture occur simultaneously. For instance, one approach may use static embeddings with a recurrent classifier, while another combines contextual Transformer representations with a different classifier. When multiple components are changed together, improvements in performance cannot be reliably attributed to the architecture alone. Controlled experiments are therefore required to assess the independent contribution of representation and architecture.

B. Limited Validation Across Personality Frameworks

Existing studies do not consistently evaluate the same architectures across different personality frameworks and data sources. Much of the reported work focuses on either **MBTI or Big Five** using specific forms of textual data, such as essays or social-media content. Consequently, it remains unclear whether an architecture that performs well in one prediction setting will maintain similar performance when the personality framework, writing style, or source dataset changes. Cross-framework and cross-dataset evaluation is therefore necessary to assess model generalizability more reliably.

C. Insufficient Analysis of Data Efficiency

Another important gap concerns the relationship between **labelled data availability and model performance**. Personality datasets can require considerable effort for data collection, annotation, and validation. However, existing comparisons do not consistently examine how different architectures perform when the amount of labelled training data is reduced. Models that maintain competitive performance with fewer labelled samples may offer important practical advantages. Therefore, data efficiency should be evaluated alongside predictive performance.

D. Inconsistent Treatment of Class Imbalance

Personality datasets may contain unequal numbers of samples across personality categories or traits. Such imbalance can cause models to favour majority classes and can make overall accuracy an unreliable indicator of performance. Although techniques such as class weighting, focal loss, and oversampling have been investigated, their use is not consistent across existing comparative studies. This variation can affect reported results and makes direct comparison between approaches more difficult. A standardized strategy for handling class imbalance is therefore required.

E. Limited Consideration of Computational and Deployment Requirements

Existing research frequently emphasizes predictive performance while giving comparatively less attention to **model size, training time, memory consumption, inference speed, and hardware requirements**. These factors are important when personality-prediction models are intended for practical deployment or resource-constrained environments. A model with slightly higher predictive performance may not be the most suitable option if its computational requirements are substantially greater.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

Architecture evaluation should therefore consider both predictive effectiveness and practical resource requirements.

F. Need for a Unified Comparative Evaluation

The limitations identified above demonstrate the need for a **unified comparative framework** in which different personality-prediction architectures can be evaluated under more consistent experimental conditions. Such a framework should consider not only accuracy, but also precision, recall, F1-score, computational complexity, data requirements, data efficiency, interpretability, and deployment suitability.

Accordingly, this study proposes a structured framework that organizes the comparison across common stages of **data acquisition, preprocessing, representation extraction, model training, validation, and evaluation**. This approach is intended to provide a more consistent basis for comparing CNN, recurrent, hybrid CNN-RNN, and Transformer-based architectures and for identifying the trade-offs between predictive capability and practical deployment requirements.

VII. PROPOSED COMPARATIVE FRAMEWORK

7.1 Overview

The proposed framework provides a systematic approach for comparing deep learning architectures used in text-based personality prediction. The framework is organized into five major stages: **data acquisition, preprocessing, representation extraction, model training, and evaluation**. Applying a common workflow across the candidate models enables a more consistent comparison of their predictive capabilities and practical requirements.

Rather than evaluating architectures only according to classification accuracy, the framework considers multiple factors, including representation strategy, architectural complexity, computational requirements, data efficiency, interpretability, and deployment suitability. This broader evaluation is intended to identify the strengths and limitations of each architecture under comparable experimental conditions.

7.2 Data Acquisition and Preprocessing

The framework supports text-based datasets commonly used for personality prediction and can also accommodate multimodal datasets when additional modalities are available. Candidate datasets include the **Essays/Big-Five dataset, MBTI Kaggle corpus, PANDORA/MBTI9k Reddit corpus, and myPersonality dataset where access is available**. Multimodal datasets containing text, audio, and video may also be considered for extended evaluation.

Before model training, preprocessing is applied to improve the consistency of the input data. Depending on the dataset, preprocessing may include normalization of text, removal of unnecessary punctuation and URLs, and tokenization. However, the framework avoids unnecessarily aggressive text cleaning because deep learning and pretrained language models can benefit from retaining meaningful linguistic and contextual information. To maintain fairness, the same preprocessing principles should be applied across the models being compared unless a specific architecture requires a representation-specific transformation.

7.3 Embedding and Representation Extraction

The framework evaluates three major representation strategies: **static embeddings, contextual embeddings, and hybrid representations**.

7.3.1 Static Word Embeddings

Static embeddings represent individual words using fixed numerical vectors. A word therefore maintains the same representation regardless of the sentence in which it appears. These embeddings can provide a relatively simple input representation for CNN and recurrent architectures while requiring comparatively lower computational resources.

7.3.2 Contextual Embeddings

Contextual embeddings generate representations according to the surrounding linguistic context. Consequently, the representation of a word can change depending on its usage within a sentence or document. Pretrained language models can either be used as fixed feature extractors or fine-tuned for the personality-prediction task. This representation family is particularly relevant to Transformer-based approaches because it enables the model to capture contextual relationships that may be difficult to represent using static embeddings.

7.3.3 Hybrid Representations

Hybrid representations combine learned textual representations with additional linguistic information. Examples include **LIWC categories, sentiment scores, and lexicon-based features**. These additional features can provide information about writing style, emotional expression, and language usage, potentially improving robustness when processing noisy or naturally generated text. The comparison of these representation families is important because embedding quality may have a substantial influence on model performance. Therefore, representation strategy should be examined separately from the downstream classification architecture wherever possible.

TABLE I. COMPARISON OF REPRESENTATION FAMILIES FOR DEEP PERSONALITY PREDICTION

Representation	Context-Aware	Typical Use
Word2Vec / GloVe (static)	No	CNN, LSTM, BiLSTM, GRU input layer
BERT / RoBERTa (contextual)	Yes (bidirectional)	Fine-tuned classifier or frozen-feature extractor
Hybrid (embedding + LIWC/lexicon)	Partial	Improves robustness on noisy social text

7.4 Candidate Deep Learning Architectures

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

The framework considers six major architecture families for comparative evaluation:
CNN – extracts informative local patterns from textual representations.
LSTM – uses gated recurrent processing to preserve relevant information across longer sequences.
BiLSTM – processes sequences in both directions to incorporate information from preceding and following context.
GRU – provides recurrent sequence modelling using a comparatively simpler gating mechanism.
Hybrid CNN-RNN – combines convolutional feature extraction with recurrent sequence modelling.
Transformer-based models – use self-attention to model relationships between different parts of the input sequence and generate contextual representations.
These architectures provide different trade-offs between representation capability, computational complexity, and predictive performance. Evaluating them within a common framework makes it possible to examine whether improvements in prediction justify additional architectural and computational complexity.

7.5 Model Training and Validation

To ensure a fair comparison, candidate models should use the **same training, validation, and test partitions** wherever the architecture permits. Possible dataset divisions include **70/15/15 or 80/10/10**, while k-fold cross-validation may be considered for smaller datasets. Training conditions should remain consistent across experiments as far as possible. The framework also incorporates early stopping based on validation performance to reduce overfitting and prevent unnecessary training. For pretrained Transformer models, the evaluation can distinguish between **frozen-feature extraction and full fine-tuning**, since these approaches have different computational requirements and adaptation capabilities.

7.6 Class Imbalance Handling

Personality datasets may contain unequal distributions among personality categories or traits. If this imbalance is ignored, a model may favour majority classes and produce misleadingly high overall accuracy.
The framework therefore considers consistent imbalance-handling strategies such as:class-weighted loss, weighted cross-entropy, focal loss, and oversampling for severely underrepresented classes. The selected strategy should be applied consistently across comparative experiments wherever possible. This helps ensure that differences in model performance are less likely to arise from differences in class-balancing procedures.

7.7 Hyperparameter and Architecture Search

Each architecture is tuned using a manageable search space appropriate to its structure. Relevant parameters may include the **number of layers, hidden dimensions, convolutional filters, learning rate, batch size, dropout rate, and number of training epochs**. For pretrained models, the framework also considers whether the underlying embeddings remain frozen or are fine-tuned.
The framework also records the amount of experimentation and computational effort required for tuning. This is important because a model requiring extensive optimization may be less practical than another model achieving comparable performance with simpler configurations.

7.8 Evaluation Metrics

Model performance is evaluated using multiple complementary metrics rather than relying exclusively on accuracy. The primary predictive measures are:
Accuracy – overall proportion of correctly classified instances.
Precision – proportion of predicted positive instances that are correctly classified.
Recall – proportion of relevant instances correctly identified.
F1-score – harmonic balance between precision and recall.
For imbalanced datasets, **macro-F1** receives particular importance because it gives equal consideration to individual classes rather than allowing larger classes to dominate the overall score. In addition to predictive performance, the framework records **training time, GPU memory consumption, and number of trainable parameters**. These measures provide an indication of the computational cost associated with each architecture.

7.9 Reproducibility and Implementation Considerations

Reproducibility is treated as an important component of the proposed framework. Details of each experiment, including model configuration, training conditions, and evaluation results, should be systematically recorded. Random seeds should be fixed where possible for data shuffling and model initialization to reduce unnecessary variation between experimental runs. The final comparison should therefore report not only which architecture achieves the highest predictive performance, but also **how much data, computation, tuning, and model complexity are required to achieve that performance**. This allows the selection of an architecture according to the requirements of a particular personality-prediction application rather than assuming that the most accurate model is automatically the most suitable one.

VIII. SYSTEM FLOW AND OPERATION

The proposed system follows a step-by-step process that converts raw input into a final personality prediction. First, text or multimodal data is collected and preprocessed through tokenisation, casing normalisation, and truncation or padding. The processed text is then passed to the embedding module, which converts tokens into numerical representations using static embeddings such as Word2Vec or GloVe, or contextual models such as BERT and

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

RoBERTa. The dataset is divided into training, validation, and test sets. Each of the six candidate architectures is trained independently using the same data and evaluation conditions. Early stopping is applied to reduce overfitting. After training, the models are evaluated on unseen test data using accuracy, precision, recall, and F1-score. The results are then compared based on both prediction performance and computational cost. The pipeline is designed to be flexible, allowing different embeddings or input modalities such as audio and video to be included without changing the complete system [10]. To maintain fairness, all models use the same data splits or cross-validation folds and the same class-imbalance strategy.

Deployment and Prediction

After evaluation, the model offering the best balance between performance and complexity can be selected for deployment. The trained model is connected to a prediction interface or API that accepts new text. The same preprocessing and embedding steps used during training are applied to new inputs. For Big Five prediction, the system can generate scores for individual personality traits. For MBTI, it can predict the relevant personality dimensions with confidence values. Since model outputs are not always perfectly calibrated probabilities, techniques such as temperature scaling can be used when reliable probability estimates are required.

Monitoring and Model Updating

Deployment is not the final stage because language patterns can change over time, especially when data comes from social-media platforms. This may cause data or language drift and gradually reduce model performance.
The deployed model should therefore be evaluated periodically using newly labelled data. If performance falls below a predefined level, the model can be retrained with recent data. Retraining requirements should also be considered when selecting an architecture. Lightweight models such as CNN or BiLSTM with frozen embeddings generally require fewer resources, while fully fine-tuned Transformers require more GPU memory and training time. Thus, although a Transformer may provide higher accuracy, a lighter model may be more suitable when frequent retraining, limited resources, or faster updates are required. Overall, the system connects data collection, preprocessing, representation, training, evaluation, deployment, and monitoring into a single flexible framework for both research and practical personality-prediction applications.

IX. COMPARATIVE EVALUATION ALGORITHM

The proposed framework compares six deep learning architectures: CNN, LSTM, BiLSTM, GRU, Hybrid CNN-RNN, and Transformer. All models use the same data partitions, preprocessing procedure, and class-imbalance strategy to ensure a fair comparison.

Let the set of candidate architectures be

$$\mathcal{A} = \{C, N, N, L, S, \begin{matrix} TMBiLSTMGRUCNN - \\ RNNTransformer \end{matrix}\}. \quad (1)$$

The input text is converted into numerical representations using the selected embedding method:

$$Z = (X), \quad (2)$$

where Xrepresents the input text and Zdenotes the resulting representation.

Each architecture is trained using the training set and evaluated on the unseen test set. The main evaluation measures are accuracy, precision, recall, and F1-score. These are defined as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (3)$$

$$Precision = \frac{TP}{TP + FP}, \; Recall = \frac{TP}{TP + FN}, \quad (4)$$

$$F1 = 2\frac{Precision \times Recall}{Precision + Recall}. \quad (5)$$

For imbalanced datasets, macro-F1 is used to give equal importance to all classes:

$$F1_{macro} = \frac{1}{K}\sum_{k=1}^{K} F\,1_k, \quad (6)$$

where $K$is the number of classes.

In addition to prediction performance, the framework considers model parameters, training time, GPU memory usage, and interpretability. The complexity of an architecture $A$is represented as

$$C_A = (P_A, T_A, G_A, I_A). \quad (7)$$

The final architecture is selected according to the requirements of the deployment environment rather than accuracy alone:

$$A^* = \arg\max_{A \in \mathcal{A}} (A \mid D), \quad (8)$$

where $S(A \mid D)$represents the suitability of architecture Aunder deployment condition D. This allows the framework to identify the highest-accuracy, best-balanced, or simplest model depending on the application requirements.

X. MATHEMATICAL FORMULATION OF CANDIDATE MODELS

A. Convolutional Neural Network

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

Given an input sequence of word embeddings x1,...,xn with xi in R^d, a convolutional filter of width k produces a feature

$$c_i = (W \cdot x_{[i:i+k-1]} + b),$$

where W is a learned weight matrix, b a bias term, and f a non-linearity such as ReLU. Feature maps from multiple filters of different widths are max-pooled over the sequence dimension to produce a fixed-length document representation, which is passed to a dense softmax layer for classification. The number of filters, filter widths, and pooling strategy are the principal hyperparameters.

B. Long Short-Term Memory

An LSTM cell updates a hidden state ht and cell state ct at each time step using input, forget, and output gates:

$$\begin{aligned} i_t &= (W_i \cdot [h_{t-1}, x_t] + b_i) \\ f_t &= (W_f \cdot [h_{t-1}, x_t] + b_f) \\ o_t &= (W_o \cdot [h_{t-1}, x_t] + b_o) \\ \bar{c}_t &= \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \bar{c}_t \\ h_t &= o_t \odot \tanh(c_t) \end{aligned}$$

where sigma is the logistic sigmoid and $\odot$ denotes element-wise multiplication. The gating mechanism allows the network to retain or discard information across long sequences, mitigating the vanishing-gradient problem that limits a simple RNN.

C. Bidirectional LSTM

A BiLSTM runs two independent LSTM layers over the input sequence, one processing tokens left-to-right (producing forward states h→t) and one right-to-left (producing backward states h←t), and concatenates them at each position: $h_t = [h_t; h_t]$ This allows the representation at any position to be informed by both preceding and following context, which is particularly useful for personality-relevant cues that depend on how a sentence resolves rather than only on how it begins.

D. Gated Recurrent Unit

A GRU simplifies the LSTM gating scheme into

$$\begin{aligned} z_t &= (W_z \cdot [h_{t-1}, x_t]) \\ r_t &= (W_r \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \end{aligned}$$

With no separate cell state and fewer weight matrices than an LSTM, a GRU has fewer trainable parameters and often trains faster, at the cost of marginally reduced representational capacity on tasks requiring very long-range dependencies.

E. Hybrid CNN-RNN

A hybrid architecture first applies one or more convolutional layers to the embedded sequence to extract local feature maps, then feeds the resulting feature sequence into a recurrent layer (LSTM, BiLSTM, or GRU) to model dependencies across the extracted local features rather than over raw tokens directly, before a final dense classification layer. This combination is intended to capture both short-range n-gram-like patterns (via convolution) and longer-range sequential structure (via recurrence) within a single model [4].

F. Transformer / BERT

A Transformer encoder layer computes self-attention over the input sequence:

$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$, where Q, K, and V are learned linear projections (queries, keys, and values) of the input embeddings and dk is the key dimension; multiple attention heads are computed in parallel and concatenated, allowing each token's representation to be updated as a weighted combination of every other token's representation in the same sequence [19]. BERT stacks multiple such self-attention layers and is pre-trained on large unlabelled text using masked-language-modelling and next-sentence-prediction objectives before being fine-tuned on the personality-labelled dataset, either by updating all encoder weights jointly with a classification head or by freezing the encoder and training only the head on its output embeddings [17].

G. Loss Functions and Optimisation

All six candidate architectures are trained by minimising a classification loss via gradient-based optimisation. For a binary personality dimension, the standard objective is binarycross-entropy:

$L = -\frac{1}{N}\sum_{i=1}^{N} [y_i \log(\hat{p}_i) + (1 - y_i)\log(1 - \hat{p}_i)]$, where yi is the true label and p_hat_i the predicted probability Under severe class imbalance, several of the reviewed studies replace plain cross-entropy with a focal loss $L_{\text{focal}} = -\sum_{i=1}^{1} [(1 - \hat{p}_i)^{\gamma} y_i \log(\hat{p}_i) + \hat{p}_i^{\gamma}(1 - y_i)\log(1 - \hat{p}_i)]$, where the focusing parameter gamma down-weights the loss contribution of easy, correctly classified examples and concentrates learning on the harder, typically minority-class, examples. Parameters are updated with an adaptive gradient-based optimiser, most commonly Adam or its weight-decay-decoupled variant AdamW, which maintains per-parameter estimates of the first and second moments of the gradient to adjust the effective learning rate for each parameter individually; a linear or cosine learning-rate warm-up and decay schedule is standard practice when fine-tuning a Transformer encoder, to avoid destructively large gradient updates to the pre-trained weights during the first few training steps.

H. Regularisation

Deep learning models usually have many more trainable parameters than traditional classifiers, which makes them more likely to overfit, especially when working with relatively small personality datasets. To reduce this risk, the reviewed models use several regularisation techniques.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

Dropout is applied after embedding, convolutional, and recurrent layers in most architectures to randomly deactivate a portion of the activations during training. L2 regularisation (weight decay) is also used during optimisation, particularly in the dense classification layer. In addition, early stopping is applied to all six candidate architectures in the proposed framework. Training is stopped when the validation macro-F1 score no longer improves for a fixed number of epochs.

XI. DATASETS AND EXPERIMENTAL SETUP

The proposed framework uses datasets that are commonly reported in deep-learning studies on personality prediction. These include the Essays/Big-Five dataset, containing around 2,467–2,468 self-report essays with binary OCEAN labels; the MBTI Kaggle dataset, containing approximately 8,600 users' forum posts across 16 MBTI personality types; and the PANDORA and MBTI9k Reddit datasets, which use user flairs and comments for MBTI and Big-Five prediction [12]. The historical myPersonality Facebook dataset was also widely used in earlier research, but academic access was closed in 2018 because of data-privacy concerns, making it unavailable for new studies [9]. Some studies also use multimodal datasets that combine text, audio, and video for Big-Five personality prediction [10]. Since MBTI datasets often have an uneven distribution of personality classes, stratified train, validation, and test splits or k-fold cross-validation are recommended to obtain more reliable results. For applications involving languages other than English, the choice of language representation is also important. Traditional embeddings such as Word2Vec and GloVe are generally trained separately for each language and may have limited coverage. In such cases, multilingual Transformer models such as multilingual BERT, or language-specific models such as IndoBERT, can provide better results [11]. Since these models are pre-trained on large multilingual or language-specific corpora, they already capture useful linguistic patterns and can often adapt to a new language with less labelled data than CNN or RNN models trained with randomly initialised embeddings.

TABLE VI. STATISTICS OF DATASETS USED IN THE REVIEWED LITERATURE

Dataset	Size / Labels	Framework
Essays / Big-Five	~2,467–2,468 essays, binary OCEAN	Big Five
MBTI Kaggle	~8,600 users, 16 types (imbalanced)	MBTI
PANDORA / MBTI9k	Reddit comments, flair-derived labels	MBTI, Big Five
myPersonality (historical)	Facebook posts; access closed 2018	Big Five
Multimodal (text+audio+video)	Varies by corpus	Big Five

Table VI provides an overview of the approximate size and labelling approach used for each dataset group. One clear pattern in the reviewed studies is the large difference in dataset size. Some studies use only a few thousand essays, while others work with millions of social-media comments. This difference can have a major effect on the reported results and may sometimes influence which architecture performs best more than the architecture itself. This issue is examined further in Section XII.

Preprocessing Consistency Across Studies

Preprocessing is another factor that makes direct comparison between studies difficult. Different studies use different maximum sequence lengths, ranging from a few dozen to several hundred tokens. They also differ in how they handle hashtags, user mentions, and emojis in social-media text. In some cases, multiple posts from the same user are combined into one document, while other studies classify each post separately and combine the predictions later. Sequence length is particularly important for Transformer models because their self-attention mechanism becomes more computationally expensive as the input length increases. As a result, studies that use shorter sequences to reduce GPU memory requirements may not fully capture the performance of a Transformer with a longer context. To make the comparison more consistent, the proposed framework recommends clearly defining and reporting the maximum sequence length used for each architecture. This helps reduce the effect of preprocessing choices that may otherwise influence the results.

XII. COMPARATIVE RESULTS AND DISCUSSION

Table II presents representative accuracy results reported in previous studies for the six candidate architectures using different datasets and embedding methods. The noticeable variation in results within each architecture shows that accuracy from a single study is not enough to determine which architecture is the best. Performance can change considerably depending on factors such as dataset size, class distribution, preprocessing, and the type of embeddings used.

TABLE II. REPRESENTATIVE REPORTED ACCURACY BY ARCHITECTURE (SELECTED STUDIES)

Architecture	Reported Accuracy Range	Representative Source(s)
CNN (document-level)	~55–65%	[18]
LSTM / GRU / BiLSTM	~40–75%	[2], [9]
Hybrid CNN-RNN	~60–78%	[3], [4]
LSTM + Transformer embeddings	up to ~87%	[7]
BERT + CNN	highest among tested combos	[1], [6]
Fine-tuned Transformer (BERT/RoBERTa)	~75–86%	[8], [11], [12]

Three main patterns can be observed from the results reported across the reviewed studies. **First**, models that use pre-trained contextual embeddings, either as fixed features or through fine-tuning, generally perform better than models trained from scratch with randomly initialised static embeddings on the same dataset. This is similar to the improvement observed when additional linguistic features such as TF-IDF and LIWC are used in traditional machine-learning approaches [7], [8], [20]. **Second**, hybrid models

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

that combine contextual embeddings with a lightweight CNN or recurrent layer, such as BERT-CNN and BERT-LSTM, often achieve results comparable to or better than a fully fine-tuned Transformer classifier. At the same time, they can reduce the number of parameters that need to be updated during training [1], [6], [8]. **Third**, there is no consistently superior recurrent architecture among LSTM, BiLSTM, and GRU. Their performance depends on factors such as the dataset, embedding method, and personality dimension being predicted [2], [15].

TABLE III. QUALITATIVE COMPARISON ACROSS PERSONALITY DIMENSIONS

Dimension	Architectures Typically Favoured	Notes
Introversion/Extraversion (I/E)	CNN, BiLSTM	Strong lexical cues; simpler models competitive
Sensing/Intuition (S/N)	BiLSTM, BERT-based	Benefits from longer context
Thinking/Feeling (T/F)	Fine-tuned Transformer	Weaker lexical cues; contextual embedding helps most
Judging/Perceiving (J/P)	Fine-tuned Transformer, hybrid	Hardest dimension across most studies

It is important to remember that accuracy values reported across different studies are not always directly comparable. Differences in test settings, embedding dimensions, and maximum sequence lengths can affect the reported results even when the underlying architecture remains the same. Similarly, studies that handle class imbalance using weighted loss functions or resampling may report lower overall accuracy but higher macro-F1 scores compared with studies that use the original imbalanced data [9], [20]. Therefore, the exact percentage differences in Tables II and III should be interpreted carefully. The more useful observation is the overall performance trend: contextual-embedding models generally achieve the strongest results, while basic CNN and RNN models are usually faster but show greater variation. Hybrid models provide a practical balance between performance and computational requirements.

Statistical Significance Considerations

Most studies included in Table II report only a single accuracy value and do not provide confidence intervals or statistical significance tests. Therefore, the rankings presented in this paper should be considered indicative rather than statistically proven. In studies that use repeated experiments or cross-validation, the variation between different runs or folds can sometimes be similar to the accuracy difference between two competing architectures. This is especially noticeable when comparing LSTM, BiLSTM, and GRU models. For this reason, the proposed framework recommends reporting results across multiple random seeds rather than selecting a superior architecture based on a single training run.

Sensitivity Analysis: Embedding Choice and Data Size

The reviewed literature also suggests that the choice of embedding has a strong influence on deep-learning performance. Replacing randomly initialised embeddings with pre-trained static embeddings such as Word2Vec or GloVe generally provides a moderate improvement. Moving from static embeddings to contextual Transformer-based embeddings usually produces a larger performance gain [7], [8], [12]. Dataset size also plays an important role. The smaller Essays dataset tends to produce lower and more variable results across deep-learning models than the larger MBTI Kaggle and Reddit-based datasets. This is expected because deep models with a large number of parameters require sufficient labelled data to learn reliable decision boundaries. Pre-training and transfer learning can partly reduce this limitation by providing useful representations before task-specific training [12], [20]. Overall, the findings suggest that using a strong pre-trained contextual embedding can have a greater impact on accuracy than simply replacing one CNN or recurrent architecture with another.

XIII. COMPUTATIONAL COST AND TRAINING INFRASTRUCTURE ANALYSIS

Although many studies focus mainly on accuracy, they provide limited information about the computational resources required to achieve those results. This makes it difficult to determine whether a small improvement in accuracy is worth the additional training cost. This section therefore compares the approximate computational requirements of the six candidate architectures based on their parameter sizes and reported training approaches.

TABLE VII. APPROXIMATE RELATIVE COMPUTATIONAL COST BY ARCHITECTURE

Architecture	Typical Parameter Count	Typical Hardware	Relative Training Time
CNN	Hundreds of thousands	Single GPU or CPU	Minutes
LSTM / GRU	Hundreds of thousands–few million	Single GPU	Tens of minutes
BiLSTM	~2x LSTM parameter count	Single GPU	Tens of minutes–hours
Hybrid CNN-RNN	Few million	Single GPU	Hours
Frozen-embedding Transformer head	Small head; embedding fixed	Single GPU	Tens of minutes
Fully fine-tuned Transformer	~110M (base) to 340M+ (large)	GPU with ≥16GB memory, ideally multi-GPU	Hours–many hours per run

Three practical points can be drawn from Table VII. **First**, fully fine-tuned Transformer models generally provide better accuracy, but they also require considerably more training time and memory than CNN or single-layer recurrent models. In some cases, the difference can be one or two orders of magnitude, which becomes important when models need to be retrained regularly as new labelled data becomes available. **Second**, using a frozen Transformer embedding

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

can provide a useful balance between accuracy and cost. In this approach, embeddings from models such as BERT or RoBERTa are generated once, and only a lightweight classification head is trained. This retains much of the benefit of contextual embeddings while keeping the training requirements closer to those of CNN or BiLSTM models. Therefore, it can be a practical choice when full Transformer fine-tuning is not feasible. **Third**, training cost and inference cost should be considered separately. A fully fine-tuned Transformer needs to process the complete encoder for every new prediction, while smaller CNN or GRU models can generally provide predictions with lower latency and less memory. This difference becomes particularly important in applications that need to process large numbers of resumes or social-media posts in near real time. Energy use and carbon emissions are not commonly reported in personality-prediction studies, but they generally increase with training time and model size. When the accuracy difference between two models is small, a smaller or distilled model may therefore be a more practical choice for deployment. These factors, together with interpretability, show why architecture selection should consider more than accuracy alone.

Knowledge distillation provides another way to balance accuracy and computational cost. In this approach, a smaller student model learns to reproduce the predictions of a larger fine-tuned Transformer, acting as the teacher. The resulting model can retain much of the teacher's performance while requiring fewer parameters and less computation during inference. Although this approach is still relatively limited in personality-prediction research, it is well established in broader NLP research. It could therefore be considered as an additional candidate in future versions of the proposed framework when sufficient personality-specific results become available.

XIV. EXPLAINABILITY AND INTERPRETABILITY TECHNIQUES

Interpretability was considered as a qualitative criterion in Table IV. However, the way a model can be explained differs considerably between CNNs, recurrent networks, and Transformers. This section therefore discusses the main techniques used for each architecture family. For **CNN-based classifiers**, individual convolutional filters can often be linked to specific words or n-gram patterns in the input text. The features that contribute most strongly to a prediction can therefore be highlighted, providing a relatively straightforward form of local explanation. For **recurrent models**, attention mechanisms can be added to identify tokens that receive higher importance during prediction. However, attention weights should not automatically be treated as a complete explanation of the model's decision, since they do not necessarily represent the actual causal influence of each word. For **fine-tuned Transformer models**, several post-hoc explanation methods can be used without changing the model architecture. These include visualising self-attention patterns, **LIME**, which creates a local approximation of the model by perturbing the input, and gradient-based methods such as **Integrated Gradients** and **SHAP**, which estimate the contribution of individual tokens to a prediction. Unlike Logistic Regression, where feature weights can provide a relatively direct global explanation, these methods mainly provide explanations for individual predictions and usually require additional computation [20]. This creates an important trade-off: more advanced architectures can provide higher predictive performance, but explaining their individual decisions becomes more difficult. A useful compromise is to combine a **frozen Transformer embedding with a lightweight classification head**, such as a shallow CNN or small dense network. While the Transformer representation itself remains difficult to interpret, the smaller classification layer is easier to examine. This approach therefore provides some of the contextual representation benefits of a Transformer while keeping the final prediction stage relatively simple and easier to analyse [1], [6], [8].

XV. THREATS TO VALIDITY

There are several limitations that should be considered when interpreting the findings of this review. Since deep-learning research in personality prediction is developing rapidly, the accuracy values reported in this paper should be viewed mainly as indicators of general trends rather than as a fixed ranking of the best-performing models. New studies and improved datasets may change these results over time.
Cross-study comparability: The results presented in Tables II, III, and V come from studies that use different dataset versions, preprocessing methods, train-test splits, and computing environments. Therefore, small differences in reported accuracy should not be treated as exact evidence that one model is better than another.

- **Publication and reporting bias**: Studies that introduce a new architecture and report improvements over existing methods are more likely to be published and cited. As a result, the literature may give more attention to Transformer and hybrid models while studies showing little or no improvement from newer architectures may be underrepresented.
- **Dataset generalisability:** The MBTI Kaggle and Essays datasets are among the most commonly used resources in the reviewed studies, but they represent specific populations rather than the general population. Consequently, a model that performs well on these datasets may not achieve the same results when applied to groups such as job applicants or other real-world populations without additional validation.
- **Label validity:** MBTI labels are often based on users' self-reported personality types. Such labels may reflect personal perception rather than the results of a professionally administered psychological assessment. This introduces uncertainty into the ground truth and is also a limitation noted in the traditional machine-learning studies reviewed in the companion paper [20].

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

- **Computational cost and reproducibility:** Some of the highest accuracy values reported in Table II are based on single training runs without repeated experiments, confidence intervals, or statistical testing. In addition, reproducing large Transformer models can require significant computational resources. These factors make it difficult to independently verify every reported result under exactly the same conditions.

XVI. ARCHITECTURE SELECTION WHICH DEEP MODEL IS SIMPLEST AND BEST

The performance of a personality-prediction system depends not only on the selected neural architecture but also on the representation used to encode the input text. Therefore, the architectures considered in this study are compared according to their ability to capture linguistic patterns, computational requirements, contextual modelling capability, and suitability for practical deployment.

TABLE IV. COMPLEXITY VERSUS ACCURACY

Architecture	Relative Params	Data Need	Interpretability
CNN	Low	Low–Moderate	Moderate (filter activations)
LSTM / GRU	Low–Moderate	Moderate	Low–Moderate
BiLSTM	Moderate	Moderate–High	Low–Moderate
Hybrid CNN-RNN	Moderate	Moderate–High	Low
Fine-tuned Transformer	High	Low (via pre-training)	Low (attention maps only)

A. CNN-Based Architecture

CNN-based models are effective at identifying local patterns in text through convolutional filters. They can capture informative words and short phrases while maintaining relatively low computational complexity. This makes CNNs suitable for applications where training speed, implementation simplicity, and resource efficiency are important.However, their ability to directly model long-range relationships is limited. Increasing the receptive field through deeper or multiple convolutional layers can improve contextual coverage but may also increase model complexity. Therefore, CNNs provide a useful lightweight baseline but may be less suitable when personality-related information depends strongly on distant contextual relationships.

B. LSTM-Based Architecture

LSTM networks are designed to model sequential information and retain relevant information over longer portions of a text sequence. Their gated memory mechanism allows the network to control which information should be retained or discarded during processing.For personality prediction, this sequential modelling capability can help capture patterns that depend on the ordering of words and broader textual context. However, LSTM models process sequences recurrently and can therefore require more computation than simpler CNN-based approaches. Their effectiveness can also vary depending on sequence length, dataset characteristics, and the quality of the input representation.

C. GRU-Based Architecture

GRU models provide a simpler recurrent alternative to LSTM. They use gating mechanisms to control information flow while maintaining a comparatively smaller architectural structure. This can reduce computational and parameter requirements while retaining the ability to model sequential dependencies.
The main advantage of GRU is therefore its balance between sequence modelling capability and computational efficiency. Nevertheless, there is no consistent evidence that GRU will outperform LSTM across all personality-prediction datasets. Its suitability should consequently be determined through controlled experimental comparison rather than architectural assumptions.

D. BiLSTM-Based Architecture

BiLSTM extends the conventional LSTM architecture by processing the input sequence in both forward and backward directions. This enables the model to incorporate information from both preceding and subsequent parts of the text when constructing contextual representations.
This characteristic is useful for personality prediction because linguistic indicators may depend on context occurring before or after a particular word or phrase. BiLSTM can therefore provide stronger contextual modelling than a unidirectional recurrent network. However, bidirectional processing introduces additional computational requirements and may be less efficient than simpler architectures for resource-constrained applications.

E. Hybrid CNN-RNN Architecture

Hybrid CNN-RNN models combine local feature extraction with sequential modelling. The CNN component identifies informative local patterns, while the recurrent component processes these extracted features to capture longer-range dependencies.This combination provides a useful compromise between the local pattern recognition capability of CNNs and the sequence modelling capability of recurrent networks. However, the additional components increase architectural complexity, training requirements, and parameter count. Consequently, the improvement obtained from a hybrid architecture should be evaluated against its additional computational cost.

F. Transformer-Based Architecture

Transformer-based models use self-attention to establish relationships between different parts of an input sequence. Unlike static embeddings, contextual representations produced by Transformer models can adapt according to the surrounding text. BERT- and RoBERTa-based approaches

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

can therefore capture richer contextual information and have strong potential for personality prediction, particularly when sufficient training data and computational resources are available. However, these models generally require greater computational resources than conventional CNN and recurrent architectures. Their complexity can also make interpretation and deployment more challenging.

G. Overall Architecture Comparison

The comparison shows that each architecture provides a different balance between predictive capability and computational requirements. CNNs are attractive for lightweight applications because of their relatively simple structure and efficient local feature extraction. LSTM and GRU models provide stronger sequential modelling capabilities, while BiLSTM additionally incorporates information from both directions of the text sequence.

Hybrid CNN-RNN architectures provide a combination of local and sequential feature extraction, but this benefit comes with increased architectural complexity. Transformer-based models provide the strongest contextual modelling capability among the considered architectures, but they generally require greater computational resources.

Importantly, the performance of an architecture cannot be considered independently of the representation used as its input. Static embeddings, contextual embeddings, and hybrid representations can produce different levels of information for the same downstream architecture. Therefore, a model achieving higher accuracy may not necessarily be superior if its improvement results primarily from a stronger representation rather than from the architecture itself. This supports the research gap identified in the study concerning the separation of representation quality from architecture quality.

H. Architecture Selection

The architecture selection should therefore be treated as a multi-criteria decision rather than an accuracy-only comparison. CNN can be preferred when simplicity, faster training, and low computational requirements are important. LSTM, GRU, and BiLSTM are suitable when sequential information plays a significant role. Hybrid CNN-RNN models can be considered when both local and sequential features are important. Transformer-based models are strong candidates when contextual modelling and predictive performance are prioritized and sufficient computational resources are available. Overall, the most appropriate architecture depends on the **dataset, representation strategy, available computational resources, prediction objective, interpretability requirements, and deployment environment** rather than on a single universal performance ranking.

XVII. ABLATION STUDY: CONTRIBUTION OF EMBEDDING CHOICE

To understand how much of the model's performance comes from the architecture itself and how much comes from the embedding representation, this section examines ablation-style comparisons reported in previous studies. These comparisons evaluate the same architecture with different levels of embedding sophistication, as summarised in Table V.

TABLE V. ABLATION PATTERN: EMBEDDING ENRICHMENT ACROSS STUDIES

Embedding Stage	Typical Effect on Accuracy	Source(s)
Randomly initialised embeddings	Baseline (lowest)	[2], [9]
Static pre-trained (Word2Vec/GloVe)	Moderate, consistent gain	[21], [22]
Contextual (BERT/RoBERTa), frozen	Large gain over static embeddings	[6], [11]
Contextual, fully fine-tuned	Largest gain; highest overall accuracy	[8], [12]

The ablation results suggest that the biggest improvement in accuracy comes from using a **pre-trained contextual embedding**, rather than simply changing the classifier between different CNN or recurrent architectures. This finding has an important practical implication for the recommendations in Section XII. Once a strong contextual embedding is selected, the choice between BiLSTM, hybrid CNN-RNN, and Transformer-based classification heads can be based more on factors such as training time, computational cost, and inference speed. Small differences in architecture-level accuracy may be less important than the variation caused by the choice of embeddings and the way class imbalance is handled.

XVIII. CONCLUSION AND FUTURE WORK

A. Conclusion

This study presented a comparative analysis of major deep learning architectures for text-based personality prediction, focusing on how different architectures and text representation strategies influence predictive performance and practical suitability. CNN, LSTM, GRU, BiLSTM, hybrid CNN-RNN, and Transformer-based approaches were considered from the perspectives of contextual modelling, computational requirements, data requirements, interpretability, and deployment suitability.The analysis indicates that **no single architecture can be considered universally superior** for every personality-prediction task. Model performance is influenced by several factors, including the personality framework, dataset characteristics, class distribution, preprocessing strategy, text representation, and evaluation methodology. Therefore, accuracy values reported under different experimental conditions should not be interpreted as direct evidence that one architecture will consistently outperform all others. CNN-based architectures remain useful when computational efficiency and

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

implementation simplicity are important. Their ability to identify local linguistic patterns makes them suitable for lightweight applications, although their capacity to capture long-range contextual relationships is comparatively limited. LSTM and GRU architectures provide stronger sequential modelling capabilities, while BiLSTM can incorporate contextual information from both directions of a text sequence. Hybrid CNN-RNN models provide a combination of local feature extraction and sequential modelling, but this advantage is accompanied by increased architectural complexity. Transformer-based approaches, particularly BERT- and RoBERTa-based models, provide strong contextual representations and substantial predictive potential. However, these advantages are associated with higher computational requirements, training costs, and challenges related to interpretability. The analysis further indicates that **the quality of the text representation can influence performance as strongly as, and in some cases more strongly than, the choice of downstream architecture**. This highlights the importance of evaluating embedding strategies and neural architectures separately when conducting comparative experiments. From a practical perspective, CNN can be considered appropriate when simplicity and computational efficiency are the primary requirements. BiLSTM and lightweight hybrid approaches can provide a compromise between contextual modelling and computational cost, while fully fine-tuned Transformer models are strong candidates when computational resources are sufficient and maximum predictive performance is the main objective. Overall, the findings demonstrate that architecture selection for personality prediction should be treated as a **multi-criteria decision**. Predictive accuracy should be considered together with computational cost, data requirements, robustness, interpretability, and deployment conditions. The proposed comparative framework provides a structured basis for evaluating these factors and selecting an appropriate deep learning architecture according to the requirements of a specific personality-prediction application.

B. Future Work

Several directions can be explored to improve the reliability, generalizability, and practical applicability of deep-learning-based personality prediction. First, future studies should evaluate multiple architectures under **identical experimental conditions**, including the same datasets, preprocessing procedures, training-validation-test splits, class-balancing strategies, and evaluation metrics. Repeated experiments with appropriate statistical analysis can further determine whether observed differences between models are consistent and statistically meaningful. Second, future research should investigate the **generalizability of personality-prediction models across different datasets and personality frameworks**. Evaluating models on both MBTI and Big Five tasks, as well as across essays, social-media content, and other text sources, can provide a clearer understanding of their robustness under different prediction settings. Third, further work should examine **data efficiency** by evaluating model performance under different amounts of labelled training data. Such experiments can identify architectures and representation strategies that maintain competitive performance when only limited labelled data are available. Fourth, future studies can investigate the contribution of **different representation strategies** through controlled ablation experiments. Comparing randomly initialized, static, contextual, and hybrid representations while keeping the downstream architecture fixed would help determine how much of the performance improvement is attributable to the input representation rather than the classifier architecture. Fifth, future research should give greater attention to **computational efficiency and deployment constraints**. Measurements such as training time, inference latency, memory consumption, parameter count, and hardware requirements can help identify models that provide an appropriate balance between predictive performance and resource usage. Finally, future work can explore **multimodal personality prediction** by integrating textual information with audio and visual features. Such extensions may provide richer behavioural representations, although they would also introduce additional challenges related to data alignment, privacy, computational cost, and model complexity.Overall, future research should move beyond reporting isolated accuracy improvements and focus on **reproducible, cross-dataset, data-efficient, and deployment-aware evaluation**. This would provide stronger evidence for selecting deep learning architectures that are both effective and practical for real-world personality-prediction applications.

C. Ethical and Privacy Considerations

Ethical and privacy issues should also be considered when deploying personality-prediction systems. Inferring personality from social-media activity or other unstructured text may occur without the level of explicit consent normally associated with a personality questionnaire. In addition, model predictions can be inaccurate, particularly for more difficult personality categories and traits. Therefore, applications in areas such as recruitment or clinical decision-making should use personality predictions as **supporting information rather than definitive labels**. Human review should be available for uncertain or borderline cases, and users should be informed about what data is being analysed and how the results are used. These safeguards are particularly important for highly accurate but less interpretable Transformer models. Combining such systems with appropriate explanation methods can help users understand and question model outputs rather than accepting them without review.

D. Broader Impact

The same capabilities that make deep-learning personality prediction useful can also create privacy risks. A model that can infer personality from ordinary online text could potentially be used for profiling purposes that individuals never expected or agreed to, including targeted messaging,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

undisclosed applicant screening, or the prediction of sensitive characteristics related to personality.

As model accuracy improves, these risks can also become more significant because more accurate models may make unauthorised profiling more effective. For this reason, successful deployment should not be judged by accuracy alone. **Clear use-case restrictions, audit records, human oversight, and regular bias evaluation** should accompany the technical evaluation of any personality-prediction system.

REFERENCES

[1] Frontiers in Psychology, "Deep Personality Trait Recognition: A Survey," Frontiers in Psychology, vol. 13, 2022.
[2] O. Hernandez and I. Scott, "Comparing Deep Recurrent Neural Network Architectures for Personality Trait Recognition from Text," in Proc. Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES), 2017.
[3] W. Xue et al., "Deep Learning-Based Personality Recognition from Text Posts of Online Social Networks," Applied Intelligence, vol. 48, 2018.
[4] X. Sun, B. Liu, J. Cao, J. Luo, and X. Shen, "Who Am I? Personality Detection Based on Deep Learning for Texts," in Proc. IEEE Int. Conf. Communications (ICC), 2018.
[5] Y. Mehta, N. Majumder, A. Gelbukh, and E. Cambria, "Bottom-Up and Top-Down: Predicting Personality with Psycholinguistic and Language Model Features," in Proc. IEEE Int. Conf. Data Mining (ICDM), 2020.
[6] Z. Ren et al., "A Multi-Label Personality Prediction Approach Based on Deep Learning Combining Semantic and Emotional Features from Social Media Text," 2021.
[7] "Text Based Personality Prediction from Multiple Social Media Data Sources Using Pre-Trained Language Model and Model Averaging," Journal of Big Data, vol. 8, 2021.
[8] "MBTI Personality Type Prediction Using BERT-LSTM and Deep Learning on Social Media Posts," in Proc. IEEE Conference Publication, 2024.
[9] "Personality Prediction System from Facebook Users," Procedia Computer Science, ScienceDirect, 2017.
[10] "Personality in 3D: Multimodal Deep Learning Framework for Big Five Trait Prediction," Neural Computing and Applications, Springer, 2026.
[11] R. L. Vasquez and J. Ochoa-Luna, "Transformer-Based Approaches for Personality Detection Using the MBTI Model," in Proc. XLVII Latin American Computing Conf. (CLEI), 2021.
[12] "Big Five Personality Trait Prediction Based on User Comments," Information (MDPI), vol. 16, no. 5, 2025.
[13] "Pushing on Personality Detection from Verbal Behavior: A Transformer Meets Text Contours of Psycholinguistic Features," arXiv preprint arXiv:2204.04629, 2022.
[14] M. S. Amirhosseini and H. Kazemian, "Machine Learning Approach to Personality Type Prediction Based on the Myers-Briggs Type Indicator," Multimodal Technologies and Interaction, vol. 4, no. 1, 2020.
[15] "Knowledge Graph-Enabled Text-Based Automatic Personality Prediction," arXiv preprint arXiv:2203.09103, 2022.
[16] "BIG5-TPoT: Predicting BIG Five Personality Traits, Facets, and Items Through Targeted Preselection of Texts," arXiv preprint arXiv:2511.09426, 2025.
[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint arXiv:1810.04805, 2018.
[18] N. Majumder, S. Poria, A. Gelbukh, and E. Cambria, "Deep Learning-Based Document Modeling for Personality Detection from Text," IEEE Intelligent Systems, vol. 32, no. 2, 2017.
[19] A. Vaswani et al., "Attention Is All You Need," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2017.
[20] Companion study, "Traditional Machine Learning Techniques for Personality Prediction: A Comparative Study on Algorithm Simplicity and Performance," 2026.
[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv preprint arXiv:1301.3781, 2013.
[22] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in Proc. Empirical Methods in Natural Language Processing (EMNLP), 2014.
[23] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in Proc. Empirical Methods in Natural Language Processing (EMNLP), 2014.
[24] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, 1997.
[25] K. Cho et al., "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," in Proc. Empirical Methods in Natural Language Processing (EMNLP), 2014.
[26] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in Proc. Int. Conf. Learning Representations (ICLR), 2015.
[27] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, vol. 15, 2014.
[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017.
[29] M. T. Ribeiro, S. Singh, and C. Guestrin, "„Why Should I Trust You?": Explaining the Predictions of Any Classifier," in Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, 2016.
[30] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2017.
[31] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic Attribution for Deep Networks," in Proc. Int. Conf. Machine Learning (ICML), 2017.
[32] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv preprint arXiv:1907.11692, 2019.
[33] J. W. Pennebaker and L. A. King, "Linguistic styles: Language use as an individual difference," *Journal of Personality and Social Psychology*, vol. 77, no. 6, pp. 1296–1312, 1999.
[34] F. Mairesse, M. A. Walker, M. R. Mehl, and R. K. Moore, "Using linguistic cues for the automatic recognition of personality in conversation and text," *Journal of Artificial Intelligence Research*, vol. 30, pp. 457–500, 2007.
[35] J. Oberlander and S. Nowson, "Whose thumb is on the scale? Identifying authorial influence in text," in *Proc. 3rd Int. AAAI Conf. Weblogs and Social Media (ICWSM)*, 2009.
[36] M. Gjurković and J. Šnajder, "Reddit: A gold mine for personality prediction," in *Proc. 2nd Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES)*, 2018, pp. 87–97.
[37] M. Gjurković, M. Karan, and J. Šnajder, "PANDORA Talks: Personality and demographics on Reddit," in *Proc. 8th Int. Workshop on Natural Language Processing for Social Media*, 2020.
[38] X. Zhao, Z. Tang, and S. Zhang, "Deep Personality Trait Recognition: A Survey," *Frontiers in Psychology*, vol. 13, 2022, Art. no. 839619.
[39] Y. Mehta, N. Majumder, S. Poria, and E. Cambria, "Recent trends in deep learning based personality detection," *Artificial Intelligence Review*, vol. 53, pp. 2313–2339, 2020.
[40] Z. Ren, Q. Shen, X. Diao, and H. Xu, "A sentiment-aware deep learning approach for personality detection from text," *Information Processing & Management*, vol. 58, no. 3, 2021, Art. no. 102532.
[41] S. Han, H. Huang, and Y. Tang, "Knowledge of words: An interpretable approach for personality recognition from social media," *Knowledge-Based Systems*, vol. 194, 2020, Art. no. 105550.
[42] J. Golbeck, C. Robles, M. Edmondson, and K. Turner, "Predicting personality from Twitter," in *Proc. IEEE 3rd Int. Conf. Privacy, Security, Risk and Trust and IEEE 3rd Int. Conf. Social Computing*, 2011, pp. 149–156.
[43] A. Vinciarelli and F. Mohammadi, "A survey of personality computing," *IEEE Transactions on Affective Computing*, vol. 5, no. 3, pp. 273–291, 2014.
[44] F. Celli, F. Pianesi, D. Stillwell, and M. Kosinski, "Workshop on computational personality recognition: Shared task," in *Proc. Workshop on Computational Personality Recognition*, 2013.
[45] M. R. Mehl, S. D. Gosling, and J. W. Pennebaker, "Personality in its natural habitat: Manifestations and implicit folk theories of personality in daily life," *Journal of Personality and Social Psychology*, vol. 90, no. 5, pp. 862–877, 2006.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 09 , September - 2026

[46] M. T. Pilehvar and J. Camacho-Collados, "Embeddings in natural language processing: Theory and advances in evaluation," *IEEE Intelligent Systems*, vol. 36, no. 2, pp. 125–134, 2021.

[47] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proc. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018, pp. 2227–2237.

[48] Y. Liu et al., "XLNet: Generalized autoregressive pretraining for language understanding," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[49] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "ELECTRA: Pre-training text encoders as discriminators rather than generators," in *Proc. International Conference on Learning Representations (ICLR)*, 2020.

[50] M. J. Hu, Y. Shen, and others, "LLM predicts human behavior: A BERT-based approach for conscientiousness personality trait detection from online content," *Acta Psychologica*, vol. 266, 2026, Art. no. 106832.