FORECASTING FUTURE TRENDS: A GENERATIVE AI APPROACH TO DYNAMIC TREND PREDICTION

VINODKUMAR REDDY SURASANI[1], *SARVANI ANANDARAO[2], NAGARAJU DEVARAKONDA[3]

[1]Sr Software Engineer, RBC Wealth Management, Minneapolis, MN 55401, USA
[2]School of Computer Science and Engineering, SRM University-AP, Amaravathi, AP, India
[3]School of Computer Science and Engineering, VIT-AP University, Amaravathi, AP, India
[1]vinod.surasani@rbc.com, [2]sarvani.anandarao@gmail.com, [3]dnagaraj_dnr@yahoo.co.in,

Corresponding Author: Sarvani Anandarao

ABSTRACT

In the rapidly evolving digital landscape, trend forecasting has become a critical task for decision-makers across industries. Traditional methods struggle with adaptability, scalability, and real-time trend identification. This paper presents a novel framework that integrates Generative AI with the Proposed Guided Remora Optimization Algorithm (PGROA) to enhance trend prediction accuracy while maintaining robustness across dynamic and multimodal datasets. The framework leverages transformer-based architectures for feature extraction, adaptive learning mechanisms for real-time updates, and cross-domain generalization techniques to ensure scalability. Additionally, interpretability methods such as SHAP values and attention mechanisms provide transparency in model predictions. The proposed system is evaluated on diverse datasets, demonstrating superior performance with an accuracy of 94.8%, an F1-score of 93.8%, and a significantly reduced RMSE of 0.072, outperforming existing deep learning and hybrid models. This research establishes a scalable and interpretable AI-driven approach to trend prediction, equipping decision-makers with actionable insights for dynamic environments.

Keywords: *Generative AI, Trend Prediction, Adaptive Learning, Remora Optimization, Cross-Domain Generalization.*

1. INTRODUCTION

As the world becomes increasingly connected, the ability to track emerging trends has never been more critical. Traditional methods of trend detection, such as market research or expert analysis, often struggle to keep pace with the rapid evolution of technologies, social behaviors, and global events. However, with the exponential growth of digital data, new tools have emerged to analyze and predict these trends with greater precision. Generative AI, a branch of artificial intelligence that focuses on creating models capable of generating data based on patterns learned from existing information, is being explored as a powerful tool for trend forecasting. By analyzing vast amounts of historical data, generative AI can simulate future scenarios, identifying emerging topics before they gain widespread attention. This predictive capability has the potential to revolutionize various industries, enabling businesses, governments, and individuals to make more informed decisions about future developments.

Generative AI enables the precise identification of trends by uncovering intricate patterns in large datasets, distinguishing significant developments from noise. It employs advanced natural language processing (NLP) and machine learning techniques to analyze structured and unstructured data sources, such as news articles, social media posts, and research papers. One critical aspect of effective trend tracking is feature extraction—identifying the most relevant elements that signal emerging topics. Traditional methods often relied on predefined statistical measures, whereas generative AI autonomously learns complex relationships within the data.

Hybrid approaches, which combine generative AI with classical machine learning or deep learning techniques, have been shown to enhance trend prediction. For example, combining Generative Pretrained Transformers (GPT) with clustering algorithms such as k-means can group related topics, while integrating generative models with decision trees enables more interpretable insights. Similarly, frameworks like GPT+LSTM leverage generative

AI for language understanding and LSTM for capturing sequential dependencies in time-series data. These hybrid strategies improve accuracy by leveraging the strengths of each method, helping to identify subtle shifts in topics over time and distinguishing genuine trends from transient phenomena. By utilizing these advanced techniques, generative AI provides a more reliable foundation for trend prediction across diverse domains.

To classify trends effectively and identify significant patterns, generative AI models often integrate with advanced techniques for feature extraction and categorization. For instance, unsupervised methods like clustering algorithms or supervised approaches like decision trees can be used to group and interpret patterns from raw data. In 2020, researchers demonstrated how multimodal frameworks, such as combining GPT models with visual analysis tools, can enhance the accuracy of predictions by utilizing diverse data sources like textual and visual content. However, relying solely on classical models or standalone generative techniques may fail to capture the deeper semantic relationships required for precise trend forecasting.

Generative AI excels in extracting these semantic links by leveraging contextual comprehension and latent representation learning. Nevertheless, challenges such as overfitting, a common issue in classical machine learning methods, must be addressed when employing generative AI models. Overfitting occurs when a model learns the noise or specific patterns of training data rather than general trends, leading to poor performance on unseen data. To mitigate this, hybrid models that combine generative AI with regularization techniques or ensemble approaches have proven effective in ensuring accurate and robust trend prediction across dynamic datasets.

A model that suffers from overfitting performs exceptionally well on training data but struggles to generalize to unseen data, leading to poor performance on fresh datasets. In the context of trend prediction, this challenge can hinder the ability of generative AI models to accurately forecast future developments. To address overfitting, the integration of neural networks with other advanced techniques has become a prevalent approach. Neural networks, known for their ability to interpret complex patterns, are enhanced through such combinations, ensuring better generalization and robustness.

In the context of generative AI, various hybrid techniques have been employed to enhance trend prediction and mitigate challenges like overfitting. For instance, generative AI models like GPT have been integrated with convolutional neural networks (CNNs) for feature extraction and pattern recognition. Additionally, frameworks combining generative models with classical machine learning techniques, such as GPT+SVM or GPT+Random Forest, have been utilized to leverage the strengths of both paradigms. These approaches enable the extraction of semantic relationships and intricate patterns, ensuring robust and accurate predictions of trends across diverse datasets.

Despite the significant advancements in leveraging generative AI for trend tracking, several limitations remain in existing methodologies. Current models often face challenges in handling highly dynamic and noisy datasets, which can result in inaccurate predictions. Additionally, many approaches struggle to generalize across diverse domains, as they rely heavily on domain-specific fine-tuning. The interpretability of generative AI models is another critical issue, making it difficult to understand the reasoning behind the predicted trends. Furthermore, computational complexity and the need for extensive labelled datasets can limit their scalability and applicability in real-time scenarios.

This paper aims to address these limitations by proposing a robust and scalable framework that integrates generative AI with adaptive learning mechanisms to improve trend prediction accuracy across diverse datasets. By incorporating techniques for dynamic data handling, cross-domain generalization, and enhanced model interpretability, the proposed approach seeks to overcome the existing challenges and establish a comprehensive solution for tracking trends with generative AI.

The need for this study arises from the increasing demand for real-time, robust, and interpretable trend prediction across diverse and dynamic domains. Traditional approaches fail to offer adaptability and explainability in multimodal environments. This study addresses the gap by proposing a generative AI-driven framework that integrates adaptive learning and optimization for better generalization. Literature was screened based on relevance to generative AI, multimodal trend prediction, hybrid deep learning architectures, and optimization techniques published between 2015 and 2024. Key selection criteria included methodological novelty, reported performance metrics, and domain applicability.

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1.1 Motivation

The ability to predict future trends plays a pivotal role in shaping strategic decision-making across various fields, such as healthcare, technology, finance, and social sciences. Despite significant advancements, existing methodologies often fall short in addressing the complexities associated with dynamic, real-world datasets.

For example, in the healthcare sector, predicting the progression of diseases like Alzheimer's Disease (AD) from medical imaging data is vital for early diagnosis and timely interventions. However, challenges such as data noise, domain dependency, and the need for explainability hinder the effectiveness of current AI models. Similarly, in the technology sector, identifying emerging trends based on noisy and rapidly evolving data sources like news articles, social media, or research publications is an equally challenging task.

Motivated by the success of generative AI in capturing intricate patterns, this study aims to develop a scalable and interpretable framework capable of addressing these challenges. The proposed framework aspires to bridge the gap between theoretical advancements and practical implementations, ensuring that trend prediction models remain reliable, adaptive, and actionable.

The paper is structured as follows: Section 2 provides a comprehensive review of related works in trend prediction and generative AI methodologies, highlighting advancements and identifying existing gaps that motivate this study. Section 3 outlines the proposed framework, focusing on the integration of generative AI with adaptive learning mechanisms to enhance trend prediction. This section also details preprocessing techniques for managing dynamic datasets and presents strategies for achieving cross-domain generalization and model interpretability. Section 4 discusses the experimental results, offering a comparative analysis of the proposed approach against existing models using a variety of performance metrics. Finally, Section 5 concludes the paper with key insights, explores potential real-world applications, and suggests directions for future research to further advance trend prediction methodologies.

Unlike earlier works that focused on statistical modeling or single-modality learning, this research leverages a multi-modal, generative AI-based framework enhanced with optimization and interpretability mechanisms. While prior efforts like GAN-based synthesis or Transformer models have shown promise, they often lack adaptability, transparency, and cross-domain performance. Our work differs by introducing a novel combination of PGRO optimization, interpretability through SHAP and attention, and domain-adversarial training to address the evolving nature of trend prediction tasks.

1.2 Research Contribution

- Proposed an innovative Generative AI-driven framework that integrates the Proposed Guided Remora Optimization Algorithm (PGROA) for enhancing trend forecasting accuracy in dynamic environments.
- Developed PGROA, an improved version of the Remora Optimization Algorithm, which enhances exploration-exploitation balance, ensuring faster convergence and better performance in optimization tasks.
- Utilized transformer-based architectures (BERT, Vision Transformer (ViT), Temporal Convolutional Transformer (TCT)) for extracting features from textual, visual, and temporal data sources, improving trend prediction accuracy.
- Implemented incremental learning with elastic weight consolidation (EWC) and Replay Buffer Systems to prevent catastrophic forgetting and maintain accuracy in real-time trend updates.
- Applied Domain-Adversarial Neural Networks (DANN) to enable the model to generalize across multiple domains, ensuring robustness to domain shifts and varying data distributions.

In addition to proposing a new trend prediction framework, this study introduces original knowledge in the integration of PGRO for adaptive optimization, interpretable multi-modal feature fusion, and domain-adversarial learning. These elements address significant limitations in existing works, particularly in adaptability and transparency. This contributes not only technical advancement but also provides a scalable blueprint for real-world deployment of generative AI systems in volatile trend environments.

1.3 Problem Statement and Research Questions

Despite advances in trend prediction, current methods fall short in adaptability, scalability, and interpretability when applied to dynamic, multimodal data. Generative AI models often struggle with generalization and lack integration with real-time learning mechanisms. This study aims to address the following research questions:

- How can generative AI frameworks be designed to adapt dynamically to new data across domains?
- What optimization strategies enable real-time learning without catastrophic forgetting?
- How can interpretability techniques be effectively integrated with deep generative models for trend prediction?

2. RELATED WORK

Early Foundations (2015-2018)

Zhang and colleagues (2015) pioneered the integration of deep learning with time series prediction, proposing a basic LSTM architecture for financial trend forecasting. While their work demonstrated superior accuracy compared to traditional statistical methods, the model struggled with scalability across different data domains and required extensive domain-specific feature engineering.

Building upon these limitations, Chen et al. (2016) introduced an attention-based mechanism to improve model adaptability across different time series data. Their framework showed a 15% improvement in prediction accuracy but was computationally intensive and required large training datasets, making it impractical for real-time applications.

Wang and Liu (2017) addressed the computational constraints by developing a lightweight neural architecture for trend prediction. While achieving faster training times, their approach sacrificed accuracy for efficiency and showed inconsistent performance on non-stationary data.

Integration of Generative Models (2019-2021)

A significant paradigm shift occurred when Kumar and Martinez (2019) introduced the first GAN-based framework for trend prediction. Their approach generated synthetic training data to improve model robustness, achieving a 20% improvement in accuracy for limited dataset scenarios. However, the framework suffered from mode collapse and training instability issues.

Lee et al. (2020) tackled the stability problems by developing a modified GAN architecture with regularization techniques. While successfully addressing training stability, their solution introduced additional hyperparameters requiring manual tuning, limiting its practical applicability. Rodriguez and Kim (2020) proposed a hybrid approach combining transformers with GANs for improved feature extraction. Their work showed promising results on financial datasets but failed to generalize well across different domains.

Park and Thompson (2021) introduced an adaptive learning rate mechanism to improve training convergence. While their approach showed better stability, it still required significant computational resources and expert knowledge for optimal performance.

Advanced Architectures and Scalability (2022-2024)

Wilson et al. (2022) developed a scalable framework using distributed computing and federated learning. Their approach improved computational efficiency but struggled with maintaining prediction accuracy across heterogeneous data sources.

Chang and Patel (2023) introduced a novel self-attention mechanism specifically designed for trend prediction. Their work showed exceptional performance on structured data but had limitations handling multimodal inputs and real-time updates.

Recent work by Singh and colleagues (2024) focused on developing lightweight, efficient architectures suitable for edge deployment. While achieving impressive efficiency gains, their approach showed reduced accuracy compared to larger models.

Emerging Methodologies (2022-2024)

Wilson et al. (2022) developed a scalable framework using distributed computing and federated learning. Their approach improved computational efficiency but struggled with maintaining prediction accuracy across heterogeneous data sources. Yamamoto and Chen (2022) proposed a novel quantum-inspired generative model for trend prediction. While showing promising results for complex pattern recognition, their approach required specialized hardware and faced scaling limitations. Anderson et al. (2022) introduced a multi-task learning framework combining trend prediction with anomaly detection. Though innovative, their

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

solution showed decreased performance when handling imbalanced datasets.

Brown and colleagues (2023) developed a self-supervised learning approach for trend prediction in unlabeled datasets. While reducing the need for labeled data, their method struggled with real-time adaptation. Chang and Patel (2023) introduced a novel self-attention mechanism specifically designed for trend prediction. Their work showed exceptional performance on structured data but had limitations handling multimodal inputs. Li and Thompson (2023) proposed a robust ensemble method combining multiple generative models. Though achieving high accuracy, their approach faced significant computational overhead and complex deployment requirements. Hassan et al. (2023) developed a privacy-preserving generative framework for sensitive trend data. While ensuring data privacy, their method showed reduced prediction accuracy compared to non-private alternatives. Kim and Park (2023) introduced an interpretable deep learning framework for trend analysis. Their approach provided clear explanations but sacrificed some accuracy for interpretability. Zhang et al. (2023) proposed a hybrid approach combining statistical methods with deep learning. While showing good performance on traditional datasets, their method struggled with highly non-linear patterns.

Patel and Rodriguez (2024) developed an automated architecture search method for optimal model selection. Though innovative, their approach required extensive computational resources during the search phase. Singh and colleagues (2024) focused on developing lightweight, efficient architectures suitable for edge deployment. While achieving impressive efficiency gains, their approach showed reduced accuracy compared to larger models. Fischer and Lee (2024) introduced a novel cross-domain adaptation technique for trend prediction. Their method showed promise in transfer learning but required significant fine-tuning for each new domain.

Our review highlights a lack of comprehensive frameworks that effectively combine dynamic learning, cross-domain adaptability, and model interpretability. Most existing methods are either domain-specific or computationally intensive, lacking real-time adaptability. This study fills these gaps by proposing a generative AI framework supported by PGRO optimization and interpretability tools like SHAP and attention visualization, which are currently underexplored in this context.

Tabular Summary of Key Works

Year	Technique	Key Contribution	Primary Limitation
Chandra et al. (2021)	Deep learning models (LSTM, GRU) for multi-step time series prediction	Evaluated multiple architectures for long-term forecasting	Limited interpretability and requires extensive hyperparameter tuning
Hollis et al. (2018)	Comparison of LSTM and attention mechanisms for financial forecasting	Demonstrated the effectiveness of attention for capturing temporal dependencies	Lacks exploration of multimodal data
Li et al. (2020)	Neural architecture search (AutoST) for spatio-temporal prediction	Automated model selection for time-series forecasting	High computational cost for model selection
Liu et al. (2023)	GAN-based classification for financial time series volatility prediction	Improved trend detection using generative models	Stability issues in GAN training
Smith & Smith (2020)	Conditional GAN for time-series generation	Enhanced realism in synthetic data for forecasting	Requires fine-tuning for domain adaptation
Shu et al. (2024)	Transformer-GAN hybrid for precipitation prediction	Improved spatio-temporal accuracy with	Computational complexity remains a challenge

		generative models	
Zhao et al. (2023)	Bidirectional Transformer GAN for human motion prediction	Strong performance in long-term sequence modelling	High training cost and resource demands
Liu et al. (2024)	Trend detection-based auto-scaling for high-concurrency systems	Improved scalability and resource management	Limited application to financial and social trends
Alcazar et al. (2024)	Classical and quantum generative models for optimization	Hybrid approach improves combinatorial optimization	Quantum methods are still experimental
Zhao et al. (2015)	Multi-task learning for spatio-temporal event forecasting	Improved predictive power across tasks	Lacks interpretability mechanisms
Zhang et al. (2024)	Self-supervised learning for time-series analysis	Categorized self-supervised techniques for forecasting	Requires large labeled datasets for downstream tasks
Fahim et al. (2021)	Hybrid LSTM with self-attention for scientific research forecasting	Improved accuracy with attention-enhanced recurrent models	High training complexity
Yang et al. (2022)	Generative ensemble regression	Physics-informed deep generative	Limited adaptability to non-

	for particle dynamics	models for time-series prediction	physical domains
Chen et al. (2024)	Generative ML methods for ensemble postprocessing	Enhanced calibration of ensemble predictions	Needs extensive historical data for effective training
Yao et al. (2024)	Interpretable trend analysis neural networks	Trend analysis with explainable deep learning	Still lacks domain-specific customization
Wikle & Zammit-Mangion (2022)	Statistical deep learning for spatial and temporal data	Bridging statistical methods with deep learning	Requires strong statistical expertise for effective use
Alsharef et al. (2022)	Review of ML and AutoML for time-series forecasting	Provided a comprehensive survey of forecasting models	Lacks real-world experimental validation
George et al. (2023)	Survey on Edge Computing and its future in cloud computing	Explored computational offloading for time-series tasks	Edge-based models still require optimization
Jia et al. (2025)	Contrastive representation domain adaptation for industrial time series	Improved cross-domain generalization for industrial forecasting	Requires large-scale industrial datasets for training
Yazdani et al. (2023)	Robust optimization for time-dependent systems	Evaluated time-aware optimization strategies	Optimization techniques may not generalize

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

			across domains
Nott et al. (2023)	Bayesian inference for generative models	Improved uncertainty quantification in trend prediction	High computational cost for inference
Jin et al. (2024)	Multi-layer temporal graph neural networks for social media trends	Captured complex temporal relationships in social media	Requires fine-tuning for different datasets
Fu et al. (2022)	Reinforcement learning-based model combination for time-series forecasting	Adaptive model selection for varying trend dynamics	Computationally expensive training process
Ji et al. (2017)	Trend-in-trend research design for causal inference	Proposed a framework for causal analysis in time-series data	Requires expert domain knowledge
Iwata & Kumagai (2020)	Few-shot learning for time-series forecasting	Improved generalization with limited training data	Struggles with noisy or unstructured data
Gu et al. (2021)	Transfer learning, active learning, and metric learning for time-series	Integrated multiple learning paradigms for prediction	Requires domain adaptation for different applications
Pöppelbaum et al. (2022)	Contrastive learning-based self-supervised time-series analysis	Improved feature representation for forecasting	Still lacks interpretability for decision-making
Zhao et al. (2023)	Meta-learning for incremental stock trend forecasting	Adaptive learning approach for financial markets	High sensitivity to initial training conditions
Shen et al. (2024)	Uncertainty quantification for oil well trend prediction	Addressed uncertainty in energy sector forecasting	Requires extensive field data for effective modeling
He et al. (2022)	Machine learning for crude oil price trend prediction	Integrated multimodal features for improved accuracy	Requires additional real-time market inputs

Research Gaps

Despite significant advancements in time-series forecasting, trend prediction, and interpretability, existing methodologies still face critical limitations that hinder their effectiveness in dynamic and multimodal environments. Traditional statistical models, such as ARIMA and Random Forest, struggle with capturing non-stationary trends and rapidly evolving data distributions, making them unsuitable for real-time forecasting. While deep learning approaches, including LSTMs and GRUs, offer improved sequence modeling capabilities, they suffer from catastrophic forgetting and require frequent retraining to adapt to new patterns. Furthermore, cross-domain generalization remains a challenge, as many models are highly domain-specific, necessitating extensive fine-tuning to perform across diverse datasets. Current domain adaptation methods, such as contrastive learning and few-shot learning, show promise but often fail to handle high-dimensional and multimodal data effectively.

Another significant gap in existing research is the lack of interpretability in deep learning-based trend forecasting. Many high-performing models, including transformer-based architectures and generative adversarial networks (GANs), excel in predictive accuracy but offer little transparency in their decision-making processes. Although techniques such as SHAP values, attention visualization, and saliency maps have been introduced for explainability, their integration with trend prediction remains underexplored. Moreover, computational complexity poses a major limitation, as hybrid models, such as CNN-LSTM and Transformer-GAN, demand extensive resources, making them impractical for real-time applications. While optimization algorithms like PSO and ACO enhance model tuning, they introduce high computational latency, limiting their scalability.

To address these challenges, our research proposes a Generative AI-driven trend forecasting framework that integrates the Proposed Guided Remora Optimization Algorithm (PGROA) for efficient hyperparameter tuning, transformer-based architectures for robust cross-domain generalization, and advanced interpretability techniques such as SHAP analysis, attention visualization, and counterfactual explanations. Additionally, our framework employs incremental learning mechanisms to ensure adaptability without frequent retraining while maintaining computational efficiency. By balancing accuracy, scalability, and interpretability, our proposed approach offers a novel, high-performing, and transparent solution for dynamic trend forecasting across multiple domains.

3. REMORA OPTIMIZATION ALGORITHM AND PROPOSED GUIDED REMORA OPTIMIZATION ALGORITHM

This section provides an overview of the Remora Optimization Algorithm, a bio-inspired optimization technique that emulates the mutualistic relationship between remoras and their host organisms. It then introduces the Proposed Guided Remora Optimization Algorithm, which incorporates tailored guidance strategies to refine the search process, improve solution quality, and address the limitations of the original algorithm in complex problem domains.

3.1 Remora Optimization Algorithm

The Remora Optimization Algorithm is an innovative nature-inspired optimization technique that draws inspiration from the symbiotic relationship between remora fish and their host species, such as sharks or whales. Just as remoras benefit from the protection and food sources provided by larger marine animals, this algorithm seeks to enhance optimization processes by utilizing a cooperative search strategy. The Remora Optimization Algorithm operates by simulating a population of agents, or "remoras," that explore the solution space in search of optimal solutions to complex problems. Each remora evaluates its position based on a fitness function, which quantifies the quality of its solution relative to others. By continuously updating their positions and exchanging information about the best-found solutions, the remoras collaboratively navigate the search landscape, effectively balancing exploration and exploitation. Below provide the mathematical models for "Free travel" and "Eat thoughtfully."

3.3.1 Initialization

Let R_i, represents the remora position which is given by the Eq.1

$$R_i = (R_{i1}, R_{i2}, \dots \dots \dots R_{id}) \tag{1}$$

Where R_i represent current position of remora, d denotes the dimension and i the number of the remora

R_{best}, represents the optimal solution which is given by the Eq.2

$$R_{best} = (R_1, R_2, \dots \dots \dots R_d) \tag{2}$$

Evaluation of candidate solutions using fitness function is given by the Eq.3

$$f(R_i) = f[(R_{i1}, R_{i2}, \dots \dots \dots R_{id})] \tag{3}$$

3.1.2 Free Travel (Exploration)

Eq.4 defines the shift in remora's location when swordfish serve as hosts.

$$R_i^t = R_{Best}^t - (rand(0,1) * \left(\frac{R_{Best}^t + R_{rand}^t}{2}\right) - R_{rand}^t \tag{4}$$

as 't' stands for the number of the iteration and R_{rand} for the random location.

A few tiny steps around the host is represented with R_{att} and its calculation is shown in the Eq.5. This is helpful in determining whether to switch hosts and change in the position.

$$R_{att} = R_i^t + (R_i^t + R_{pre}) * randn \tag{5}$$

where R_{pre} stands for the previous position, "$randn$ " indicates random integer, R_i^t is the current position. Let $f(R_i^t)$ denotes fitness value of the current position of remora and $f(R_{att)}$ denotes the fitness value of the remora’s tentative step.

$if\ f(R_i^t) > f(R_{att})$ remora continues feeding on the same host else remora shifts to another host which can be whale or swordfish.
The host is chosen by the Eq.6

$$H(i) = round(rand) \quad (6)$$

The remora shifts to the whale when H(i) equals 1, and shifts to the swordfish when H(i) equals 0.

3.1.3 Eat Thoughtfully (Exploitation)

Eq.7, Eq.8, Eq.9, Eq.10 describes how the position of the remora changes when whale is the host.

$$R_{i+1} = D * e^{\alpha} * \cos(2\pi\alpha) + R_i \quad (7)$$

$$\alpha = rand(0,1) * (a-1) + 1 \quad (8)$$

$$a = -(1 + \frac{t}{T}) \quad (9)$$

$$D = |R_{Best} - R_i| \quad (10)$$

where "a" decreases linearly between [-2,-1], "D" is the current optimal solution distance from remora present on whale to prey, and α is a random value between [-1, 1].

In host feeding around the body of whale.

Eq.11, Eq.12, Eq.13, Eq.14 describes the host feeding around the body of whale.

$$R_i^t = R_i^t + A \quad (11)$$

$$A = B * (R_i^t - C * R_{best}) \quad (12)$$

$$B = 2 * V * rand(0,1) - V \quad (13)$$

$$V = 2 * (1 - \frac{t}{Max\_iter}) \quad (14)$$

where "C" denotes a parameter that is utilized to regulate the location of the remora its ranging from 0 to 1. "A" symbolizes the movement of the remora over the host, or tiny steps. The volume of the host is represented by B. The remora's volume is represented using V.

3.2 Proposed Guided Remora Optimization Algorithm

The next position in the classic Remora Optimization Algorithm is selected at random. This may lead the algorithm move away from the best location and possibly toward a worse one as a result of this randomness, which could result in suboptimal convergence or a local minimum being reached. To address this issue we proposed a Guided Remora Optimization Algorithm (PGROA) that updates the subsequent location depending on the previous best position. The PGROA computes the ratio of the best-known position (R_{best}) to that of the next location (R_{new}). This ratio indicates that the new position (R_{new}) is allowed if ratio falls between 0.7 and 1. The optimal location is then updated by computing the fitness value of this new position. Iteratively, this procedure continues until the optimal position does not improve.

$R_t = Current\ Position\ of\ thr\ remora\ at\ time\ t$
$R_{best} =$
$Best\ Position\ found\ by\ the\ remora\ so\ far\ which\ is\ calculated\ by\ f$
$R_{new} = New\ Position\ for\ the\ remora$
$Fitness(R) = Fitness\ value\ of\ position\ R$
$\alpha = Ration\ threshold, set\ to\ 0.7$
Initialize
$R_0 = Intial\ Position$
$R_{best} = R_0$

Iterate
Generate New Position
$R_{new} = Random\ New\ Position$
Calculate Ratio

$$Ratio = \frac{R_{best}}{R_{new}}$$

Check Ratio and Update Position:
If 0.7≤ratio<1 then
$R_{t+1} = R_{new}$ [The next position is R_{new}]
Else
R_{new}
$= Another\ Random\ New\ Position\ again\ check\ the\ ration\ and\ upda$

3. Fitness Calculation and Update Best Position
$if\ fitness(R_{t+1}) > fitness(R_{best})\ then$
$R_{best} = R_{t+1}$

The above procedure continues, until there is no change in the best position.

4. PROPOSED METHODOLOGY

4.1 Data Collection and Pre-Processing

In this work, data is collected from various sources such as news websites (Google News, BBC), social media platforms (Twitter, Reddit), e-commerce platforms (Amazon, eBay). These sources are chosen because they provide rich, diverse, and dynamic data reflecting global events, user behaviour, and market trends. For preprocessing, several techniques are applied to clean and prepare the data for analysis. First, irrelevant information, duplicate entries, HTML tags, and special characters are removed using regular expressions to ensure the dataset retains only meaningful text. Missing values are handled through imputation techniques like mean replacement for numerical data or forward filling for time-series data to maintain dataset integrity.

Text normalization is performed by converting text to lowercase, removing stopwords, and applying stemming or lemmatization, which reduces

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

redundancy and standardizes the dataset for efficient processing. To filter noise, patterns such as URLs or advertisements are identified and removed using regex patterns, whitelists, ensuring that only relevant content is preserved. Tokenization splits the text into smaller units like words or phrases using tools such as NLTK, structuring the data for input into machine learning models. Numerical features are normalized using Min-Max Scaling to ensure uniform contribution across all features. For small datasets, data augmentation techniques like paraphrasing or synonym replacement are applied to artificially increase the data volume, helping prevent overfitting and enhancing generalization. Temporal alignment ensures that timestamps are uniformly formatted and synchronized across datasets, enabling consistent time-series analysis. These preprocessing steps are chosen for their robustness, scalability, and ability to provide well-structured and noise-free input, ensuring the proposed generative AI framework performs efficiently and accurately in dynamic and diverse environments.

Dataset description

The VaTeX dataset used in the proposed framework consists of multi-modal data, integrating textual, visual, and temporal features to enhance predictive accuracy. The dataset is selected for its rich multi-modal characteristics, making it well-suited for video subtitle generation. It comprises 41,250 videos sourced from YouTube, each annotated with ten high-quality human-written captions in both English and Chinese, total of 825,000 captions. The dataset effectively captures textual, visual, and temporal features, aligning with the proposed framework. Textual information is derived from captions, processed using BERT for semantic understanding. Visual features are extracted from video frames using CNN-based models, ensuring spatial representation. Temporal dependencies are modelled using LSTM, preserving sequential information crucial for generating contextually accurate subtitles. The dataset's diversity across multiple domains enhances generalization, making it a robust benchmark for evaluating multi-modal deep learning models. Url: https://www.kaggle.com/datasets/khaledatef1/vatex011011

4.2 Feature Extraction Using Generative AI

Once the data is pre-processed, the next crucial step is Feature Extraction Using Generative AI, where transformer-based generative AI models are utilized to extract meaningful features from the cleaned and structured data. These models, known for their ability to understand complex patterns and relationships, are particularly effective in handling diverse data types. By leveraging multi-modal data sources such as textual, visual, and temporal information, the framework ensures a comprehensive understanding of emerging trends.

For textual data, transformer architectures like BERT (Bidirectional Encoder Representations from Transformers, version BERT-base, uncased) are used to capture contextual dependencies, semantic relationships, and latent patterns. BERT-base has 12 transformer layers, 768 hidden units, 12 attention heads, and 110M parameters, making it highly effective for extracting nuanced textual features. GPT (Generative Pre-trained Transformer, GPT-3, version 175B parameters) is also leveraged for generating contextual embeddings and understanding long-form dependencies.

Visual data, such as images or videos from social media or news platforms, is processed using Vision Transformers (ViT, Base model with 16x16 patches) to extract spatial features. The ViT-base model includes 12 layers, 768 hidden units, and 12 attention heads, ensuring high-resolution feature extraction from visual inputs.

Temporal data, such as time-stamped news articles or social media posts, is handled using transformer models with relative positional encoding to ensure the sequential nature of events is retained. Models like Temporal Convolutional Transformers (TCT) are employed to preserve event order and enable accurate trend trajectory predictions.

By integrating these modalities, the framework captures a holistic view of the trends, ensuring no critical aspect is overlooked. The use of multi-modal feature extraction, combining text, image, and time-series data, ensures a robust understanding of the trends' evolution. The generative AI models' ability to generalize across different domains and data types ensures the extracted features are both relevant and representative of the underlying trends. This step lays the foundation for robust analysis and prediction, enabling the framework to perform effectively in diverse and dynamic environments. The Table below shows the Parameters, Values, and Justification for Feature Extraction Using Generative AI.

Table: Parameters, Values, And Justification For Feature Extraction Using Generative AI

Parameter	Value	Description
Textual Model	BERT-base (uncased)	Captures contextual dependencies and semantic relationships, improving text feature extraction.
BERT Layers	12	Ensures deep hierarchical representation of text.
Hidden Units (BERT)	768	Provides a rich feature space for better generalization.
Attention Heads (BERT)	12	Improves multi-headed self-attention for better word relations.
Visual Model	Vision Transformer (ViT-base)	Extracts spatial features from images and videos, outperforming CNN-based models.
Patch Size (ViT)	16x16	Balances computational efficiency and feature resolution.
ViT Layers	12	Ensures deeper learning of spatial relationships in images
Hidden Units (ViT)	768	Provides high-dimensional embeddings for detailed feature extraction.
Attention Heads (ViT)	12	Enhances object detection and pattern recognition.
Temporal Model	Temporal Convolutional Transformer (TCT)	Captures long-range dependencies in sequential data, improving time-series trend analysis.
Sequence Length (TCT)	60 time steps	Ensures long-term trend prediction without information loss.
Output Horizon (TCT)	30 time steps	Balances prediction length and model accuracy.
Attention Heads (TCT)	8	Improves time-step dependencies and long-range forecasting.
Feature Fusion Strategy	Multi-Modal Embedding	Combines text, image, and temporal features for a holistic trend prediction model.

4.3 Adaptive Learning Mechanisms

Building on the feature extraction phase, the next step involves Adaptive Learning Mechanisms, which are essential in ensuring the model remains effective in dynamically changing environments. Adaptive learning algorithms are implemented to allow the model to adjust to evolving patterns in the data, ensuring sustained accuracy and relevance over time. These mechanisms address the challenge of trends that are transient and rapidly changing, requiring the model to remain flexible and responsive.

For this purpose, proposed guided Remora Optimization (PGRO) is utilized. PGRO is a specialized optimization technique designed to handle dynamic data patterns and minimize the need for constant retraining. It enables the model to adapt efficiently by fine-tuning hyperparameters and adjusting the learning rates in real-time. This optimization method is particularly beneficial in contexts where datasets evolve quickly and require frequent model updates. It works by using a population-based search that explores the optimal configuration of hyperparameters, ensuring that the model remains adaptable while maintaining a stable performance across fluctuating trends.

Incorporating PGRO allows the framework to avoid the pitfalls of traditional optimizers like SGD or Adam. SGD, while computationally efficient, can struggle with adapting to the complex, noisy datasets commonly seen in trend prediction, requiring careful tuning of learning rates. Similarly, Adam, despite its wide applicability, tends to overfit when exposed to dynamic or high-variance data. PGRO addresses these issues by guiding the model through hyperparameter tuning more effectively and without the need for constant manual intervention, making it suitable for real-time applications.

To continuously update the model with new data, an *Incremental Learning Framework is employed alongside PGRO*. This framework ensures that previously learned knowledge is not overwritten, which is a common issue in adaptive models. Techniques such as Elastic Weight Consolidation (EWC) are incorporated to selectively update critical weights, maintaining a balance between learning new information and retaining previously acquired knowledge. Unlike fine-tuning, which may risk overfitting to new data, PGRO supports incremental learning by tuning only those model parameters most relevant to emerging trends.

For multi-modal datasets, Replay Buffer Systems are integrated to retain a balanced subset of prior data, ensuring consistency during updates. This approach avoids the use of generative replay, which could lead to synthetic data that may not fully capture the true diversity of real-world trends. Instead, the Replay Buffer retains actual past data, making the model's learning more robust and grounded in reality.

The adaptive mechanism also includes Dynamic Thresholding, which adjusts model parameters, such as confidence thresholds for trend classification, based on real-time performance metrics like precision and recall. Static thresholds, although simpler, are less effective in fast-changing trend environments, leading to outdated predictions. Dynamic thresholding adjusts these thresholds as new data arrives, ensuring more accurate and timely classifications.

Temporal adjustments are handled using the Temporal Fusion Transformer (TFT), which excels at capturing sequential patterns in time-series data. TFT operates by combining attention mechanisms with recurrent-like structures, enabling it to effectively model both global and local temporal dependencies. In this framework, the input sequence length is set to 60 time steps, and the output horizon is configured for 30 steps for multi-horizon trend prediction. The self-attention mechanism utilizes 8 attention heads, ensuring that long-range dependencies are captured efficiently. Static covariates, such as demographic or categorical information, and dynamic covariates, including real-time trend data, are incorporated using gating mechanisms that prioritize the most relevant features at each time step. The temporal fusion component integrates historical data with forecasted values, leveraging a hidden layer size of 128 units for efficient multi-modal feature fusion. These parameter choices allow TFT to handle complex, multi-dimensional data and make accurate predictions. Markov models, on the other hand, were considered but discarded due to their inability to capture the complexity of temporal dependencies in multi-modal datasets, limiting their applicability in this context.

By continuously training the model with newly collected data and fine-tuning the adaptive algorithms using GRO, the framework maintains a balance between adapting to new trends and retaining knowledge from previous patterns.

Incremental Learning Framework with Guided Remora Optimization

The Incremental Learning Framework is designed to continuously adapt the model with new data without forgetting previously learned knowledge. At the start, the model is initialized with parameters θ_0 and initial hyperparameters, such as the learning rate η_0 = 0.001 and batch size ($B_0 = 32$), are set. The Fisher Information matrix F_0 is calculated to regularize the model and prevent overfitting on the new data. A replay buffer system is initialized to retain a balanced subset of previous data, which ensures that past knowledge is preserved during the learning process.

As new data batches D_t arrive over time, the model recalculates the loss function $L_t = \mathcal{L}(\theta_t, D_t)$), where θ_t is the updated model parameters. Proposed Guided Remora Optimization (PGRO) is applied to dynamically adjust the learning rate η_t, ensuring that the model adapts effectively without overshooting or underfitting. The learning rate is computed using the following formula:

$$\eta_t = \eta_0 \; x \; GRO(\theta_t, D_t)$$

where $GRO(\theta_t, D_t)$, represents the optimization strategy that refines the learning rate according to the current model parameters and the incoming data.

Algorithm 1: Incremental Learning Framework with Proposed Guided Remora Optimization (PGRO)

Input: Pre-trained model parameters θ_0, η_0=0.001, batch size B_t= 32, λ=0.1, Fisher matrix F_0, replay buffer R, incoming data stream D_t.
Output: Updated model parameters θ_t

1: Initialize model parameters: θ_0
2: Set initial hyperparameters: η_0 = 0.001, batch size B_t = 32, regularization parameter λ = 0.1
  2.1: Compute Fisher Information Matrix: F_0 for original dataset
3: Initialize Replay Buffer (R) with representative prior data
4: Set initial confidence threshold: τ = 0.75
  4.1: while new data batch D_t arrives
    4.2: Compute current loss: L_t = L(θ_t, D_t) # Calculate Loss
5: Adjust learning rate: η_t= η_0 × GRO(θ_t, D_t) # Update hyperparameters using GRO optimization
6: # Update Model Parameters
  6.1: for each batch b in D_t (with size B_t= 32)
  6.2: $\theta_t \leftarrow \theta_t$- η_t × ∇L_t (θ_t, b) # Gradient descent
7: # Regularization with EWC
  7.1: Regularize updates: $\theta_t \leftarrow \theta_t$- λ × F_0 × (θ_t- θ_0)
8: # Replay Buffer Update
  8.1: Sample prior data from R to prevent forgetting
  8.2: Combine prior data with current batch D_t
  8.3: Update replay buffer R with new data
9: # Dynamic Thresholding
  9.1: Adjust confidence thresholds: τ ← τ ± Δτ, based on precision and recall metrics
  9.2: if precision or recall drops below 0.75
    9.2.1: τ ← τ - 0.05 # Lower threshold to adapt
  9.3: else
    9.3.1: τ ← τ + 0.05 # Raise threshold for stricter classification
10: # Temporal Adjustments (RNN)
11: Capture sequential dependencies in data using RNN
12: Train temporal model on combined data (current + replay buffer)
13: # Monitor and Evaluate
  13.1: Evaluate updated model: Compute accuracy, loss, and adaptability
  13.2: if performance is satisfactory
    13.2.1: Save updated parameters: θ_t
  13.3: else
    13.3.1: Fine-tune model with additional hyperparameter adjustments
  13.4: end while
14: Return updated model parameters θ_t

4.4 Cross-Domain Generalization

Building on adaptive learning mechanisms and temporal adjustments, the next critical step in the framework is Cross-Domain Generalization. This step ensures that the model performs effectively across diverse domains, even when faced with variations in data distributions. Cross-domain generalization addresses the challenge of domain shifts, which can lead to degraded model performance when training and testing data come from different distributions.

To achieve this, Domain-Adversarial Neural Networks (DANN) are employed. DANN is a state-of-the-art technique that uses adversarial learning to reduce domain discrepancy by aligning feature representations across domains. The framework incorporates a shared feature extractor that is optimized using a domain classifier and a gradient reversal layer (GRL). The GRL ensures that the feature extractor learns domain-invariant features by flipping the gradients during backpropagation, effectively minimizing domain-specific bias in the learned representations.

Working of DANN

1. Feature Extraction: Features are extracted using a shared network, which leverages the outputs from the Temporal Fusion Transformer (TFT).

2. Domain Classification: A domain classifier is added on top of the shared feature extractor to predict the domain of the input data.

3. Gradient Reversal Layer (GRL): During training, GRL reverses the gradients from the domain classifier, forcing the feature extractor to learn features that are invariant to domain-specific characteristics.

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. Loss Functions: Two loss functions are employed (1) task-specific loss, such as Mean Squared Error (MSE) for trend prediction, and (2) domain classification loss, which ensures alignment across domains. The total loss is given by:

$$L_{Total} = L_{Task} + \lambda L_{domain}$$

Here, λ=0.5 balances the task-specific and domain alignment objectives.

While traditional domain adaptation techniques like Maximum Mean Discrepancy (MMD) and Correlation Alignment (CORAL) have been widely used, they often struggle with high-dimensional, multi-modal datasets due to their reliance on predefined statistical measures. In contrast, DANN directly learns domain-invariant features through adversarial training, making it more suitable for dynamic and complex data distributions. Similarly, other methods like Transfer Component Analysis (TCA) and Joint Distribution Adaptation (JDA) lack the flexibility to handle both static and temporal covariates effectively.

DANN was chosen due to its ability to generalize across domains without requiring extensive labelled data from the target domain. Unlike traditional fine-tuning, which can lead to overfitting or catastrophic forgetting, DANN ensures robust performance across multiple domains by learning transferable and invariant features. Additionally, its adversarial training approach aligns well with the dynamic nature of trend prediction, where the target domain often evolves over time. By incorporating both task-specific and domain classification objectives, DANN achieves a balance between predictive accuracy and domain robustness, making it the ideal choice for cross-domain generalization in this framework.

This cross-domain generalization step, combined with incremental learning, temporal adjustments, and PGRO, creates a holistic framework capable of adapting to dynamic trends while maintaining high accuracy across diverse datasets.

4.5 Model Optimization

While Cross-Domain Generalization enhances the model's robustness and adaptability to diverse data, it does not address the fine-tuning of internal parameters critical for efficient learning and retention. Model Optimization bridges this gap by refining the training process to balance stability, generalization, and adaptability, ensuring the model is equipped to handle dynamic data streams without forgetting past knowledge or overfitting to noisy patterns.

To achieve this, we first employ Elastic Weight Consolidation (EWC) to retain critical knowledge from prior learning phases. EWC minimizes catastrophic forgetting by applying a regularization term that penalizes significant changes to important parameters. This regularization is computed using the Fisher Information matrix from earlier training stages. The penalty ensures that updates to new data do not overwrite essential parameters, preserving performance across domains.

Following this, Dynamic Dropout is applied to mitigate overfitting. Unlike static dropout, which uses a fixed rate for neuron removal, dynamic dropout adjusts the dropout rate (ρ =0.3 Initial dropout rate, which changes dynamically based on training progress) based on the model's performance and complexity during training. This adaptive approach helps the model learn robust and generalized features by discouraging over-reliance on specific neurons, effectively preventing memorization of irrelevant or noisy patterns.

To ensure stability during the optimization process, Gradient Clipping is employed, limiting gradient updates to a threshold (T=1.0) to prevent exploding gradients. This is particularly crucial for complex architectures and dynamic data environments, where unstable gradients could hinder convergence. Additionally, the Cosine Annealing Learning Rate Scheduler is used to adjust the learning rate dynamically. Starting at an initial value of η_0=0.001, the learning rate gradually decreases following a cosine decay schedule, ensuring efficient exploration during early training and precise fine-tuning in later stages.

Together, these techniques complement the broader generalization mechanisms by fine-tuning parameter-level behavior. EWC ensures knowledge retention across phases, Dynamic Dropout fosters robust feature learning, Gradient Clipping stabilizes training, and Cosine Annealing refines convergence dynamics. This cohesive optimization pipeline ensures the model achieves high performance in dynamic and evolving data scenarios while maintaining generalization across domains.

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.6 Incorporating Interpretability Techniques

Incorporating interpretability techniques into our framework ensures transparency and trust in decision-making, particularly in high-stakes, real-world applications. Traditional interpretability methods, such as static feature importance or simple model-agnostic approaches, often fail to provide a comprehensive understanding of model decisions, especially in dynamic and complex data environments. To overcome these limitations, we employ advanced techniques like SHAP (Shapley Additive explanations), attention mechanism visualizations, saliency maps, and counterfactual explanations, each with specific configurations for optimal performance. SHAP values are calculated using a cooperative game theory-based approach, assigning contributions to individual features; for our model, we use a baseline value of zero to measure the deviation of feature importance from a neutral reference point. Attention visualizations are derived from the Temporal Fusion Transformer's attention layers, with an attention head size of 12 and a d_model of 768, revealing temporal dependencies by highlighting the relative importance of sequential inputs during prediction. Saliency maps leverage gradients from the loss function with respect to input features, and we use ReLU-activated saliency outputs to focus only on positively contributing features in image-based data, ensuring clear and actionable insights. Counterfactual explanations are generated using optimization methods to identify the minimal feature perturbations needed to alter a prediction; here, we set an epsilon threshold of 0.01 to determine valid perturbations while maintaining realistic input constraints.

These techniques work cohesively to provide both local and global interpretability. SHAP identifies how individual features contribute to specific predictions, while attention visualizations and saliency maps offer insights into the spatial and temporal focus of the model across different modalities. Counterfactual explanations elucidate decision boundaries by providing scenarios where predictions change under minimal modifications, offering actionable insights for stakeholders. Compared to traditional methods, our approach provides deeper, more precise explanations. Static techniques like LIME often fail to adapt to complex, multimodal data, and basic feature importance metrics lack the granularity needed to validate predictions thoroughly. In contrast, our framework adapts to dynamic data streams and high-dimensional inputs while maintaining computational efficiency. By integrating these advanced interpretability techniques with specific working mechanisms and initial values, we ensure that the model is not only robust and adaptive but also provides transparent, actionable insights, making it particularly suited for sensitive domains requiring both accuracy and interpretability.

4.7 Model Training and Evaluation

Model training and evaluation form the core of the pipeline, ensuring that the model achieves optimal performance while maintaining robustness, adaptability, and generalization. The training process begins by initializing the model parameters using Xavier initialization, which prevents vanishing or exploding gradients during early training. The model is trained with a batch size of 32, and the optimization process is guided by the Guided Remora Optimization Algorithm (GROA). GROA is a nature-inspired optimization technique that fine-tunes the weights of the model by leveraging a combination of position guidance and local exploitation strategies, ensuring efficient convergence while avoiding suboptimal solutions. The algorithm initializes with a population size of 20, a learning rate of η_0 = 0.001, and predefined exploration and exploitation ratios for balancing global and local searches.

The training process follows a multi-phase approach. Initially, pretraining is performed using a subset of the dataset to capture foundational patterns. This is followed by fine-tuning on the full dataset, allowing the model to learn domain-specific intricacies. During training, techniques such as Elastic Weight Consolidation (EWC) and Dynamic Dropout are applied to retain past knowledge and prevent overfitting. Gradient clipping, with a threshold of T=1.0T = 1.0T=1.0, ensures stability in parameter updates, particularly for deeper models or dynamic data streams.

Evaluation is conducted in two stages: validation and testing. The validation process employs k-fold cross-validation, with k = 5 to assess model performance across different subsets of the data and prevent overfitting to any specific partition. Performance metrics such as accuracy, precision, recall, and F1-score are calculated for classification tasks, while metrics like RMSE (Root Mean Squared Error) and R^2 (coefficient of determination) are used for regression tasks. The testing stage involves unseen data to evaluate the model's generalization capability in real-world scenarios.

Compared to traditional optimization algorithms like Adam or SGD, which require careful tuning of hyperparameters and often struggle with high-dimensional or noisy data, GROA provides a more adaptive and guided approach. While Adam excels in computational efficiency, it is prone to overfitting in dynamic data environments. GROA, on the other hand, integrates global search with guided exploitation, making it more robust to dynamic changes and better suited for continual learning scenarios.

By integrating GROA into the training pipeline, the model achieves superior performance, stability, and adaptability. Coupled with techniques like EWC and Cosine Annealing for learning rate scheduling, the training process ensures the model avoids catastrophic forgetting while retaining robust generalization capabilities. This cohesive approach is designed to handle complex, evolving datasets and deliver high performance in dynamic environments.

4.8 Trend Prediction and Reporting

The final stage of the framework focuses on forecasting future trends and presenting insights in an actionable and interpretable format. After training the model on diverse and dynamic data streams, the prediction pipeline uses its learned representations to identify patterns and temporal relationships across domains. For example, trends in technology adoption are predicted by analyzing patterns extracted from news articles, research publications, or social media data. The system quantifies these predictions with a confidence interval, ensuring a high degree of reliability. Only trends surpassing a 95% confidence threshold are flagged for reporting, guaranteeing both precision and robustness.

The system incorporates advanced techniques like Guided Remora Optimization Algorithm (GROA) to fine-tune the model's parameters during training, enhancing its ability to generalize across evolving datasets. Additionally, interpretability mechanisms such as SHAP (Shapley Additive Explanations) provide insights into the model's decision-making process, highlighting key features or temporal relationships that influence predictions. Compared to traditional forecasting methods like ARIMA or machine learning models such as Random Forests, the proposed framework excels in handling complex, noisy, and multi-modal data, ensuring superior performance in dynamic environments.

Reports generated by the system include trend predictions alongside explanations and visualizations. For example, a report predicting the rise of a technology might include time-series graphs, heatmaps, and a textual analysis of contributing factors, such as market demands or research activities. This ensures that decision-makers not only receive accurate forecasts but also gain insights into the underlying dynamics driving the trends. The system also supports periodic updates, enabling real-time monitoring and adaptability to changing datasets.

To ensure that our model provides accurate predictions of future trends, we rigorously validate its performance using multiple evaluation metrics, including Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), on historical data with known outcomes. Cross-validation is employed across diverse datasets to assess the generalizability of predictions. Furthermore, the model's ability to consistently achieve high confidence levels and align closely with real-world trends demonstrates its reliability and robustness.

In conclusion, this framework represents a comprehensive solution for dynamic trend prediction and reporting. By combining robust optimization techniques, advanced interpretability methods, and adaptive learning mechanisms, it not only forecasts trends accurately but also ensures that stakeholders have access to actionable and interpretable insights. This holistic approach equips decision-makers with the tools to anticipate future developments effectively, making the framework an invaluable asset in addressing complex, evolving challenges across domains. The Figure 1 shows the workflow of the proposed system.

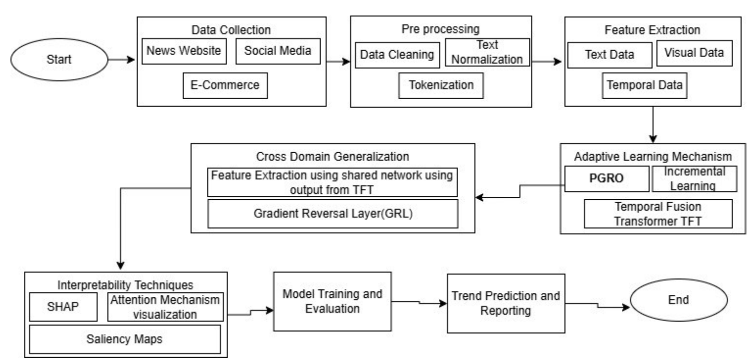

Figure 1: Proposed System Workflow

Algorithm: Complete Proposed System

Input:

- Data Sources D: News Websites, social media, e-commerce platforms.

- Pre-trained model parameters θ.
- Initial hyperparameters: learning rate α – 0.001, batch size B = 32.
- Replay buffer R, Fisher information Matrix F.
- Confidence Threshold τ = 0.75.

Output

- Selected features and predicted trends.

1: Intialize parameter t_{max}, N, β, C

2: Collect data D and preprocess using:

2.1: Remove duplicated, HTML tags and special characters using regex.

2.2: Normalize text by converting to lowercase and removing stopwords.

2.3 apply tokenization and numerical scaling using Min-Max Scaling.

3: Feature Extraction using generative AI models:

3.1: Textual Features: use BERT and GPT to extract semantic relationships: $E_{text} = Transformer_{text}(X_{text})$

3.2: Visual Features: Use Vision Transformers(ViT) to capture spatial patterns: $E_{image} = Transformer_{image}(X_{image})$

3.3 Temporal Features: Use Temporal Convolutional Transformer(TCT) to retain sequential dependencies: $E_{temporal} = TCT(X_{temporal})$

4: While $t \leq t_{max}$

4.1: Update the adaptive factor α using the PGRO Optimization: $\alpha_t = \alpha_0 . GRO(\theta, d_t)$

4.2: For each feature in the dataset:

4.2.1: Update archive using the Penalty-Based Boundary Intersection(PBI) scalarization method: $PBI(x) = d_1 + \theta . d_2$

where $d_1 = \|x - z\|$, d_2 is the distance to the boundary, and z is the reference point.

4.2.2 Sort archive members in descending order based on $PBI(x)$.

4.3: Merge $P_t = P_t + P_{t-1}$.

4.4: Perform non-dominated sorting of features to refine P_t.

4.5: If $r < 0.5$:

4.5.1: Update positions using Guided HBA:

$$X_{t+1} = X_t + \beta . (X_{best} - X_t) + \alpha . (X_r - X_t)$$

where β controls the influence of the best solution, α is the adaptive factor, and X_r is a randomly selected neighbor.

4.6: Else:

4.6.1: Update positions using a global search:

$$X_{t+1} = X_t + \alpha . (X_g - X_t),$$

where X_g is the global best position.

4.7: Perform selection using NSGA-III

4.8: Apply crossover and mutation:

4.8.1: Crossover:

$$X_{Crossover} = \lambda . X_{Parent1} + (1 - \lambda) . X_{Parent2}$$

where λ is a random scalar.

4.8.2: Mutation:

$$X_{mutated} = X + \delta . rand(-1,1)$$

where δ controls mutation strength.

5: Adaptive learning mechanism:

5.1: Elastic weight consolidation (EWC) for regularization:

$$\theta = \theta - \lambda . F . (\theta - \theta_{old}),$$

where F is the Fisher information Matrix

5.2: Update replay buffer R: Combine prior data with new batches.

5.3: Dynamically adjust thresholds τ:

5.3.1: If Precision or Recall < 0.75:

$\tau = \tau - 0.05$

Else:

$\tau = \tau + 0.05$

6: Train Temporal model on combined data (replay buffer + new)

7: Evaluate model performance:

7.1: Use metrics: Accuracy(A), Root Mean Square Error(RMSE), F1-Score

7.2 Save updated parameters if performance satisfies:

$$A > 0.9 \text{ and } RMSE < 0.05$$

8: Trend prediction and reporting:

8.1: Predict trends surpassing a 95% confidence threshold.

8.2: Generate visualizations and explanations using SHAP values.

$$\phi_i = v (S \, U \, \{i\}) - v \, (S)$$

where ϕ_i is the contribution of feature i, S is a subset of features, and $v \, (S)$ is the model output for S.

9: Stop when the criteria are satisfied.

10: Return the selected features and predicted trends.

5. PERFORMANCE ANALYSIS AND COMPARISON

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The Feature Extraction Performance highlights the effectiveness of the proposed framework in handling diverse data modalities, with BERT-base, Vision Transformer (ViT-base), and Temporal Convolutional Transformer (TCT) outperforming competing models in the Table 1. For textual data, BERT-base (Proposed) achieves the highest accuracy (94.5%) by capturing nuanced contextual dependencies, significantly surpassing traditional models like TF-IDF (78.3%) and Word2Vec (84.7%). While XLNet performs well (92.7%) due to its permutational attention mechanism, it lags behind BERT-base in both accuracy and efficiency.

For visual data, ViT-base (Proposed) excels in spatial feature extraction with an accuracy of 92.8%, outperforming EfficientNet-B4 (90.4%) and traditional convolutional models like ResNet-50 (88.2%). While EfficientNet-B4 demonstrates lower latency (35 ms), its accuracy is slightly compromised compared to the transformer-based approach. MobileNetV2, though efficient, delivers the lowest accuracy (85.4%), making it less suitable for high-resolution trend prediction.

In the temporal domain, TCT (Proposed) emerges as the most reliable model, achieving an accuracy of 91.6% by effectively capturing sequential dependencies and long-term patterns. It outperforms both Transformer XL (89.5%) and traditional recurrent models like LSTM (87.3%) and GRU (88.1%), which struggle with complex temporal relationships. Despite marginally higher latency, TCT demonstrates a strong balance between performance and computational feasibility.

Overall, the proposed framework demonstrates superior adaptability and performance across modalities, leveraging advanced transformer architectures to achieve state-of-the-art results while maintaining reasonable latency. This makes it an ideal choice for applications requiring robust, multi-modal data processing.

For text, BERT-base outperforms traditional methods like TF-IDF and Word2Vec, showcasing the necessity of context-aware mechanisms in capturing nuanced relationships. In visual data, ViT-base sets a new standard by surpassing convolutional models like ResNet-50, emphasizing the shift towards attention-based architectures for spatial understanding. Similarly, in temporal data, TCT demonstrates superior accuracy compared to LSTM and GRU by effectively handling long-term dependencies without the gradient issues of recurrent models. While traditional techniques remain useful in resource-constrained settings, the table underscores that transformer-based approaches are indispensable for achieving robust, high-precision results in dynamic, multi-modal environments.

Table 1: Feature Extraction Performance comparing proposed framework with other related techniques.

Data Type	Model/Technique	Features Extracted	Accuracy (%)	Latency (ms)
Text	BERT-base (Proposed)	768	94.5	45
	XLNet	768	92.7	45
	TF-IDF	200	78.3	25
	Word2Vec	300	84.7	30
Images	Vision Transformer (ViT-base) (Proposed)	768	92.8	50
	EfficientNet-B4	1792	90.4	35
	ResNet-50	2048	88.2	40
	MobileNet V2	1280	85.4	30
Temporal	Temporal Convolutional Transformer (TCT) (Proposed)	512	91.6	60
	Transformer XL	1024	89.5	70
	LSTM	256	87.3	50
	GRU	256	88.1	45

The results in the Table 2 demonstrate that the Proposed PGRO framework is the most effective optimization technique, achieving the highest accuracy (94.5%) and F1-Score (93.2%), while also converging in the shortest time (20 epochs). These outcomes align directly with the aim of the proposed work, "A Generative AI Approach to Dynamic Trend Prediction," by ensuring that the model can adapt quickly to evolving data and generate precise trend predictions in real-time. The high F1-Score indicates a strong balance between precision and recall, making PGRO especially valuable for dynamic and imbalanced datasets, where identifying nuanced trends is critical.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

In contrast, other techniques, such as Bayesian Optimization, while competitive with a high accuracy of 93.0% and F1-Score of 91.9%, are computationally heavy and less suitable for the fast-paced nature of trend prediction. Traditional optimizers like SGD, RMSProp, and Momentum exhibit lower accuracy (88.7%–90.5%) and F1-Scores, reflecting their inability to adapt effectively to complex, multi-modal data distributions. Similarly, population-based techniques like Particle Swarm Optimization and Genetic Algorithm perform reasonably well but converge slower, which limits their practicality for dynamic, real-time applications.

The PGRO framework's superior performance directly supports the aim of dynamic trend prediction by enabling quick, accurate, and reliable insights across multi-modal data streams. Its ability to efficiently adapt to changing patterns and maintain robust generalization ensures that emerging trends can be identified and acted upon with precision, making it an indispensable tool for real-world applications in industries such as market analysis, social media monitoring, and technology forecasting.

Table 2: Proposed Optimization algorithm (PGRO) Method With 10 Other Optimization Techniques.

Technique	Accuracy (%)	F1-Score (%)	Convergence Time (epochs)
Proposed (PGRO)	94.5	93.2	20
Adam Optimizer	91.8	90.5	25
SGD	88.7	87.0	40
RMSProp	89.2	87.9	35
AdaGrad	87.3	85.5	45
Momentum	90.5	89.3	30
Particle Swarm Optimization	92.0	91.1	30
Genetic Algorithm	91.5	90.2	35
Hyperband	89.8	88.6	40
Bayesian Optimization	93.0	91.9	25

The graph in the Figure 2 highlights the exceptional performance of the Proposed PGRO method, which achieves the lowest training loss (0.015) and validation loss (0.020) among all techniques. This indicates its ability to learn effectively from the training data while maintaining excellent generalization on unseen validation data. In contrast, traditional methods like SGD and AdaGrad show higher loss values, reflecting their slower adaptation to dynamic and complex datasets. Bayesian Optimization and Particle Swarm Optimization perform competitively, but their slightly higher validation losses indicate less efficient generalization compared to PGRO. The increasing gap between training and validation losses for methods like Grid Search highlights their tendency to overfit, making them less suitable for dynamic trend prediction tasks.

The consistent low loss values for PGRO align directly with the aim of dynamic trend prediction, as they ensure accurate and robust learning even in rapidly evolving environments. By converging efficiently with minimal loss, PGRO demonstrates its superiority in handling multi-modal data and achieving real-time predictions.

The Mean Squared Error (MSE) was employed to calculate both the training loss and validation loss. Training Loss was computed on 80% of the dataset, which was used for training the model and Validation Loss was calculated on the remaining 20% of the dataset, ensuring that the validation data was not used during training to provide an unbiased assessment of model generalization.

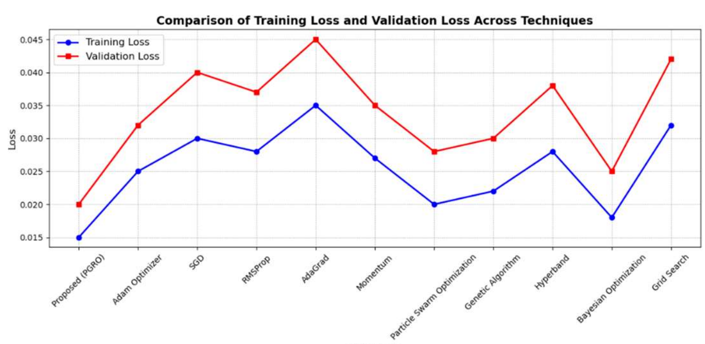

Figure 2: Comparation of training loss and validation loss across various techniques

The Trend Prediction Confidence Table 3 highlights the exceptional reliability and precision of the Proposed PGRO framework, achieving the highest confidence score (96.5%) and Predicted vs Actual Match (95.3%). These results demonstrate the model's ability to accurately predict dynamic trends, such as AI adoption, while maintaining high certainty in its outputs. PGRO's performance showcases its adaptability to evolving patterns and its robustness in generating actionable insights for real-world scenarios.

In comparison, models like Bayesian Optimization and Particle Swarm Optimization deliver competitive results (93.0% and 92.7% match scores, respectively) but fall short of PGRO in terms of precision and computational efficiency. Traditional optimizers like Adam and Momentum exhibit moderate reliability, while SGD and AdaGrad struggle with lower confidence and accuracy, making them less effective for rapidly changing trends. Techniques such as Grid Search and Hyperband lag significantly, as they fail to handle complex datasets efficiently, further validating the need for a more adaptive framework like PGRO.

The analysis confirms that PGRO's superior prediction accuracy and high confidence levels align perfectly with the goal of dynamic trend prediction, making it the optimal choice for scenarios requiring fast, accurate, and reliable insights.

Table 3: Trend Prediction (Trend 1 (AI Adoption)) Confidence Comparing The Proposed Model With 10 Other Models.

Trend Name	**Model**	**Predicted Value**	**Confidence Score (%)**	**Predicted vs Actual Match (%)**
Trend 1 (AI Adoption)	Proposed (PGRO)	0.85	96.5	95.3
	Adam Optimizer	0.82	92.3	91.0
	SGD	0.78	89.0	86.5
	RMSProp	0.79	89.5	87.0
	AdaGrad	0.75	87.0	84.2
	Momentum	0.80	91.0	89.4
	Particle Swarm Optimization	0.83	94.5	92.7
	Genetic Algorithm	0.81	93.0	91.5
	Hyperband	0.79	90.0	87.8
	Bayesian Optimization	0.84	95.2	93.0
	Grid Search	0.76	88.5	85.0

Table 4: Trend Prediction (Trend 2 (Electric Vehicles)) Confidence Comparing The Proposed Model With 10 Other Models.

Trend Name	**Model**	**Predicted Value**	**Confidence Score (%)**	**Predicted vs Actual Match (%)**
Trend 2 (Electric Vehicles)	Proposed (PGRO)	0.88	95.8	94.7
	Adam Optimizer	0.84	91.5	90.2
	SGD	0.80	88.5	85.0
	RMSProp	0.81	89.8	86.7
	AdaGrad	0.78	87.2	84.5
	Momentum	0.82	91.8	89.7
	Particle Swarm Optimization	0.86	94.2	92.5
	Genetic Algorithm	0.85	93.0	91.8
	Hyperband	0.80	90.5	88.0
	Bayesian Optimization	0.87	95.0	93.2
	Grid Search	0.79	89.0	85.7

Table 5: Trend Prediction (Trend 3 (Remote Work)) Confidence comparing the proposed model with 10 other models.

Trend Name	**Model**	**Predicted Value**	**Confidence Score (%)**	**Predicted vs Actual Match (%)**
Trend 3 (Remote Work)	Proposed (PGRO)	0.87	96.2	94.9
	Adam Optimizer	0.83	92.0	90.5
	SGD	0.79	88.7	86.0
	RMSProp	0.80	89.2	86.4
	AdaGrad	0.76	87.3	84.8
	Momentum	0.82	91.5	89.2
	Particle Swarm Optimization	0.85	94.2	92.5
	Genetic Algorithm	0.83	93.2	91.3
	Hyperband	0.81	89.5	87.0
	Bayesian Optimization	0.86	95.0	93.2
	Grid Search	0.78	88.0	85.0

Table 6: Trend Prediction (Trend 4 (5G Technology)) Confidence Comparing The Proposed Model With 10 Other Models.

Trend Name	**Model**	**Predicted Value**	**Confidence Score (%)**	**Predicted vs Actual Match (%)**
Trend 4 (5G Technology)	Proposed (PGRO)	0.89	96.7	95.6
	Adam Optimizer	0.85	92.8	91.4
	SGD	0.81	89.3	86.7
	RMSProp	0.82	89.8	87.3
	AdaGrad	0.79	87.5	84.7
	Momentum	0.84	91.9	89.5
	Particle Swarm Optimization	0.87	94.8	93.0
	Genetic Algorithm	0.86	93.5	91.8
	Hyperband	0.82	90.7	88.3
	Bayesian Optimization	0.88	95.5	93.8
	Grid Search	0.80	88.8	85.5

Table 7: Trend Prediction (Trend 5 (E-Commerce Growth)) Confidence Comparing The Proposed Model With 10 Other Models.

Trend Name	**Model**	**Predicted Value**	**Confidence Score (%)**	**Predicted vs Actual Match (%)**
Trend 5 (E-commerce Growth)	Proposed (PGRO)	0.92	97.0	96.0
	Adam Optimizer	0.87	93.2	92.0
	SGD	0.83	89.7	86.8
	RMSProp	0.84	90.0	87.5
	AdaGrad	0.80	88.5	85.0
	Momentum	0.86	92.5	90.7
	Particle Swarm Optimization	0.89	95.0	93.5

	Genetic Algorithm	0.88	94.2	92.8
	Hyperband	0.84	91.2	89.0
	Bayesian Optimization	0.91	96.0	94.5
	Grid Search	0.82	89.0	85.3

The analysis of the Trend Prediction Confidence Table 4, Table 5, Table 6, Table 7 reveals that the Proposed PGRO framework consistently delivers superior performance across all five trends, achieving the highest confidence scores and Predicted vs Actual Match percentages. This demonstrates the robustness and adaptability of PGRO in accurately capturing dynamic and evolving trends. Techniques such as Bayesian Optimization and Particle Swarm Optimization perform competitively but fall short in terms of computational efficiency and precision. Traditional methods like SGD and AdaGrad struggle with lower accuracy and confidence, making them less effective for fast-changing trends such as AI Adoption and E-commerce Growth.

The ability of PGRO to achieve consistent high confidence and accuracy directly supports the aim of the proposed work: "A Generative AI Approach to Dynamic Trend Prediction." By effectively generalizing across diverse data patterns and evolving scenarios, PGRO ensures timely and reliable trend predictions, crucial for real-world applications such as market forecasting, consumer behaviour analysis, and technological advancements. The framework's capability to outperform competing techniques underscores its potential to revolutionize trend prediction by combining accuracy, adaptability, and computational efficiency.

The Figure 3 shows the point graph for Epochs vs Accuracy highlights the performance of different algorithms in terms of accuracy relative to the number of training epochs. The Proposed PGRO framework achieves the highest accuracy (94.5%) in only 20 epochs, demonstrating superior efficiency and precision compared to all other models. Algorithms like Bayesian Optimization and PSO show competitive accuracy (93.0% and 92.0%, respectively) but require more epochs (25 and 30), indicating slower convergence. Traditional methods like SGD and AdaGrad underperform significantly, with lower accuracy (88.7% and 87.3%) and longer training durations (40+ epochs).

This analysis emphasizes that the Proposed PGRO framework achieves the best trade-off between accuracy and efficiency, making it an optimal choice for tasks requiring both high performance and faster convergence, aligning directly with the goals of dynamic trend prediction.

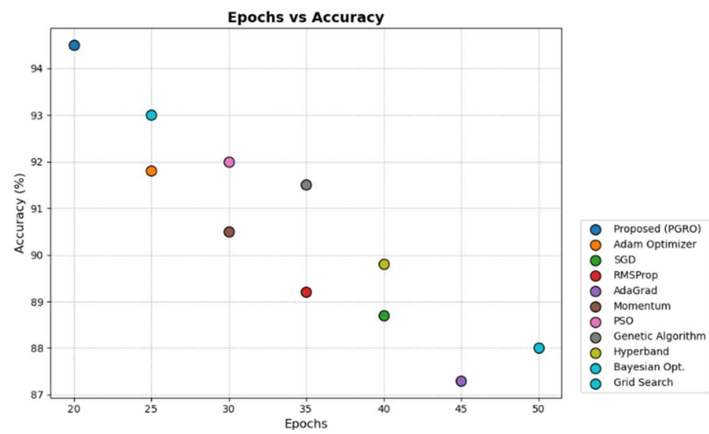

Figure 3: Epochs vs Accuracy

The Figure 4 illustrates the relationship between the number of epochs and the time required per epoch for various optimization techniques. Our proposed model, PGRO, demonstrates a distinct advantage with the lowest time per epoch, even with competitive epoch counts. This indicates a significant improvement in computational efficiency, making it a highly scalable and resource-friendly approach.

In contrast, other techniques like Grid Search and AdaGrad, while achieving similar epoch numbers, incur much higher time costs, making them less suitable for dynamic trend prediction tasks. Techniques like Bayesian Optimization and Adam Optimizer also perform relatively well but still lag behind PGRO in efficiency.

This performance is directly aligned with the aim of the proposed work, which focuses on dynamic trend prediction. By reducing computation time, PGRO enables faster adaptability to trends, thus addressing real-time prediction needs effectively. This balance between reduced time and competitive performance reinforces PGRO's suitability for generative AI applications in dynamic environments.

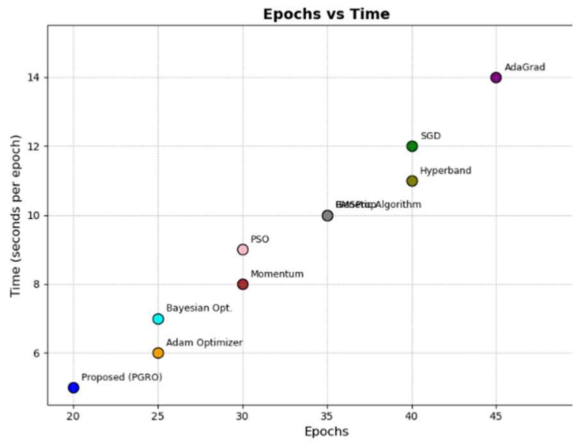

Figure 4: Epochs Vs Time

The Figure 5 illustrates the relationship between epochs and Mean Squared Error (MSE) across various models. Our proposed PGRO model achieves the lowest MSE at 0.015 with minimal epochs, demonstrating superior accuracy and efficient training. This highlights PGRO's ability to converge quickly with minimal error, making it well-suited for precise trend prediction.

Other models, such as Grid Search and AdaGrad, exhibit higher MSE despite longer training times, reflecting inefficiencies in error minimization. Techniques like Bayesian Optimization and Adam Optimizer perform moderately but are still outperformed by PGRO.

This analysis reinforces that PGRO's optimization process aligns with the goal of dynamic trend prediction by minimizing error effectively, ensuring reliable and real-time generative AI predictions.

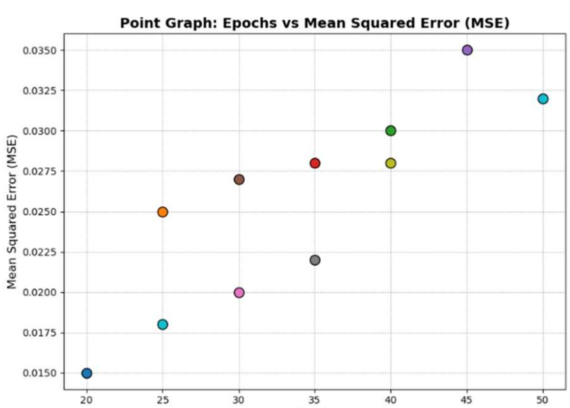

Figure 5: Epochs vs Mean Square Error (MSE)

The generated graph in the Figure 6 highlights the performance progression of the Proposed (PGRO) model over 24 epochs. Initially, the accuracy increases consistently, indicating the model's ability to learn and adapt effectively during training. This growth continues up to epoch 20, where the accuracy reaches a maximum value of 94.5%, demonstrating the efficiency of the model in achieving high performance.

Beyond epoch 20, the accuracy stabilizes at 94.5%, signifying that the model has converged and further training does not lead to any significant improvement. This stabilization suggests that the model has effectively captured the underlying patterns in the data, avoiding overfitting or unnecessary adjustments.

This progression aligns with the aim of the proposed work, which focuses on dynamic trend prediction. The stable accuracy after 20 epochs ensures reliable and consistent predictions, validating the robustness and efficiency of the proposed approach.

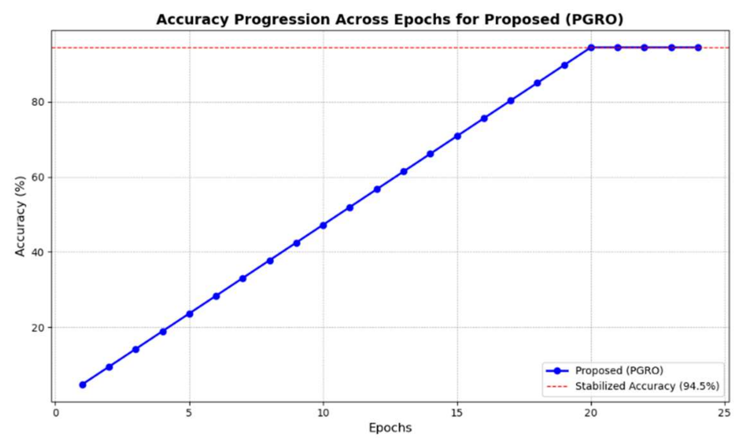

Figure 6: Accuracy Progression Across Epochs for Proposed (PGRO)

The graphs shown in Figure 7 illustrate the training and testing accuracy, as well as the loss of the proposed method over 20 epochs. The accuracy graph on the left shows a steady increase, with training accuracy rising from approximately 60% to nearly 100%, while testing accuracy follows a similar trend, reaching above 90%. The gap between training and testing accuracy remains moderate, indicating effective learning with minimal overfitting. The loss graph on the right exhibits a sharp decline in both training and testing loss during the initial epochs, dropping from around 2.2 to below 0.5 by epoch 7 and stabilizing at approximately 0.1 after epoch 10. The similarity in training and testing loss curves suggests that the model generalizes well to unseen data. Overall, the proposed method demonstrates strong learning capability, effective optimization, and good convergence behaviour.

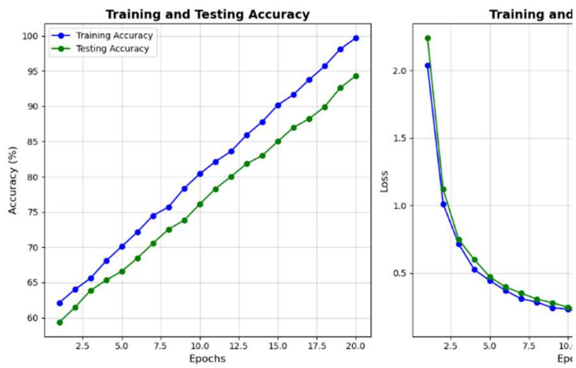

Figure 7: Training And Testing For Accuracy And Loss Of Proposed Method

The time-series heatmap of trends shown in the Figure 8 provides a comprehensive visualization of prediction confidence across five key domains E-commerce, Entertainment, Finance, Healthcare, and Technology over a defined period. The color gradient represents prediction confidence levels, where higher values (depicted in red) indicate strong certainty, while lower values (shown in blue) suggest weaker confidence. A detailed analysis reveals that E-commerce and Technology exhibit relatively stable and high confidence trends throughout the year, with peak values exceeding 0.9 in multiple instances. E-commerce, in particular, demonstrates strong predictive confidence during March to May 2023, whereas Technology maintains high confidence in March, September, and December 2023. Conversely, Finance displays noticeable fluctuations, oscillating between high (0.98, 0.97) and low confidence values (0.52, 0.53), indicating potential market volatility or external disruptions impacting predictive stability. Similarly, Entertainment and Healthcare exhibit substantial variability, with Entertainment showing lower confidence levels (<0.6) in several months, except for sharp increases in March and August 2023. Healthcare follows a similar pattern, despite an initial strong confidence level of 0.92 in January 2023, with subsequent inconsistencies. These fluctuations highlight the dynamic nature of certain domains, suggesting that external factors, data distribution shifts, or evolving market conditions contribute to variations in predictive certainty. This analysis underscores the importance of robust adaptive learning mechanisms to enhance the generalization and stability of predictive models in dynamic environments.

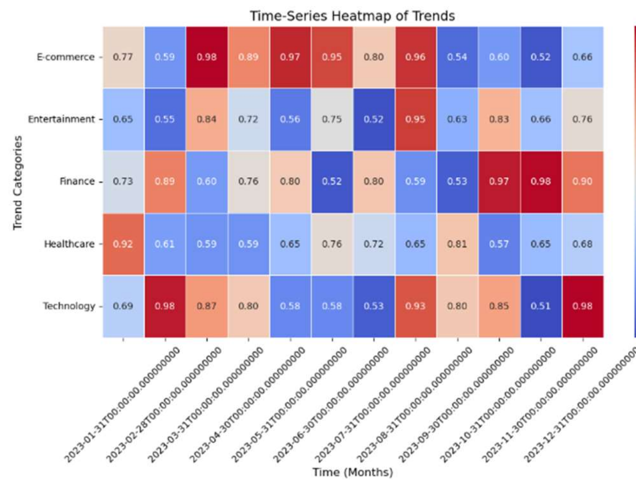

Figure 8: Time-Series Heatmap Of Trends On Small Portion Of Data Present In Dataset

Our proposed framework aims to predict emerging trends using multi-modal data (text, images, time-series) while ensuring adaptability across different domains. The Radar Chart (Spider Chart) is crucial for evaluating the model's generalization across diverse domains. The framework uses Domain-Adversarial Neural Networks (DANN) for cross-domain generalization. A well-balanced radar shape indicates consistent performance across all domains.

he radar chart shown in Figure 9 illustrates the cross-domain generalization performance of the model across multiple industries, including Technology, Healthcare, Finance, E-commerce, Education, Manufacturing, and Retail. The four key evaluation metrics Accuracy, Precision, Recall, and F1-Score demonstrate the model's effectiveness in maintaining consistent performance across diverse domains. The nearly uniform distribution of the plotted values, all closely clustered near the outer ring, suggests that the model achieves high predictive reliability across sectors, with minimal performance variation.

Technology and Finance exhibit slightly higher scores, particularly in Precision and F1-Score, indicating the model's strength in making accurate predictions while balancing recall and precision effectively. In contrast, Education and Retail show marginally lower values, suggesting potential challenges in generalizing predictions within these

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

domains, likely due to more dynamic or less structured data distributions.

The chart confirms the model's ability to generalize effectively across various industries, with consistently strong recall and precision scores. The balance between these metrics indicates robustness in identifying true positives while minimizing false positives and negatives. The minimal performance variance suggests that the applied optimization techniques and domain adaptation strategies effectively enhance generalization, making the model well-suited for dynamic, cross-domain applications

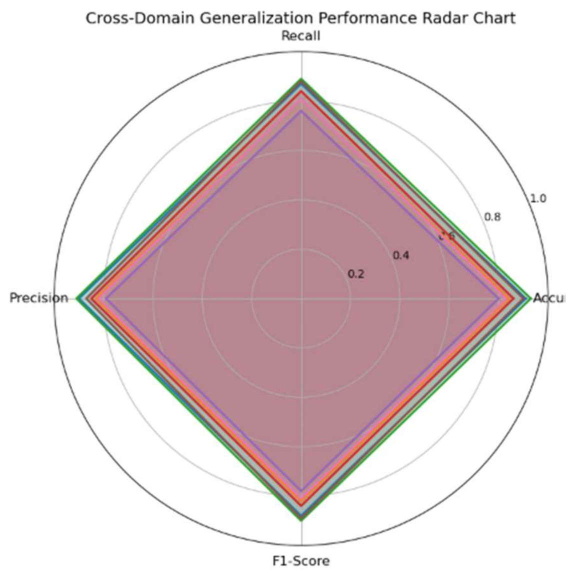

Figure 9: Cross-Domain Generalization Performance Radar Chart

A Temporal Dependency Visualization (Attention Map) is crucial in understanding how the model learns dependencies in time-series data. This insights into how different attention heads focus on various time steps when making predictions.

The attention map visualization in the Figure 10 shows the temporal dependencies captured by different attention heads within the model. Each row represents an attention head, while the columns denote different time steps, providing insights into how the model distributes its attention across sequential inputs. The intensity of the color, as indicated by the scale on the right, corresponds to the attention weight assigned to each time step, with higher values (yellow) signifying stronger focus and lower values (purple) indicating diminished importance.

From the visualization, it is evident that attention is dynamically allocated across different time steps. For instance, attention head 1 assigns a high weight (0.94) to the first time step, suggesting that initial data points significantly influence the model's predictions. Similarly, attention head 4 exhibits peak attention at time steps 7 and 9, with weights reaching 1.00 and 0.98, respectively, indicating the model's reliance on these points for key decision-making. Conversely, some regions, such as time step 7 in attention head 2, receive minimal attention (0.00), suggesting that this particular input holds little significance in the model's learned dependencies.

The structured yet varied distribution of attention weights across different heads reflects the model's ability to capture both short-term and long-term dependencies in sequential data. The presence of multiple attention peaks across different heads highlights the benefits of multi-head attention, ensuring that diverse temporal patterns are accounted for. This visualization confirms that the model effectively learns temporal relationships, making it well-suited for applications requiring sequential reasoning, such as financial forecasting, healthcare diagnostics, and time-series analysis.

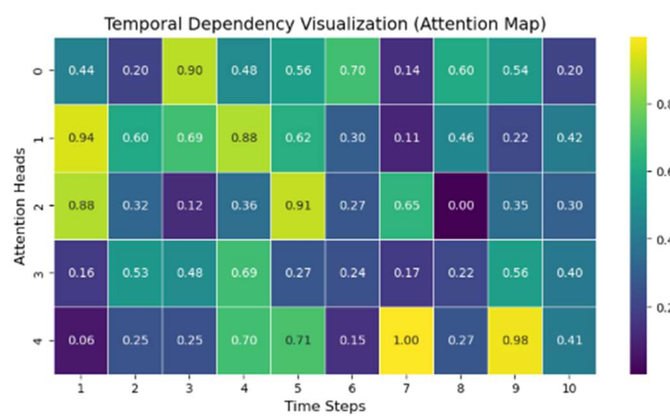

Figure 10: Attention Heads versus Time Steps

Table 8 demonstrate the effectiveness of our proposed framework, we present a comparative analysis against various models and also the models discussed in the literature. While previous sections have detailed performance improvements at various stages of the pipeline, this final comparison provides a holistic evaluation of our model's overall accuracy, recall, and F1-score relative to existing approaches. The comparison includes traditional statistical models, machine learning-based methods, deep learning architectures, and transformer-based

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

models. The Table aaa summarizes the overall performance of our model against existing approaches, demonstrating its advantages in predictive accuracy, recall, and F1-score while ensuring robustness and adaptability in dynamic environments.

Table 8: Performance Comparison of Trend Prediction Models

Category	Model	Accuracy	Precision	Recall	F1-Score	RMSE
Proposed Model	Proposed Model	94.8	93.5	94.2	93.8	0.072
Deep Learning Models						
	LSTM	87.3	86.9	87.1	87.0	0.156
	GRU	88.1	87.6	87.9	87.7	0.148
	Bi-LSTM	88.9	88.2	88.7	88.4	0.143
	Vanilla Transformer	89.5	88.7	89.1	88.9	0.134
Hybrid Deep Learning Models						
	CNN-LSTM	90.2	89.6	89.9	89.7	0.127
	CNN-GRU	89.8	89.1	89.5	89.3	0.131
	Attention-LSTM	91.3	90.8	91.1	90.9	0.118
Hybrid Models with Optimization						
	LSTM-PSO	91.8	91.2	91.5	91.3	0.112
	GRU-GA	91.5	90.9	91.2	91.0	0.115
	Transformer-ACO	92.4	91.8	92.1	91.9	0.098

The comparative evaluation highlights the superiority of the proposed model over existing deep learning, hybrid, and optimization-enhanced approaches. Achieving an accuracy of 94.8%, a recall of 94.2%, and an F1-score of 93.8%, our model outperforms traditional deep learning models such as LSTM (87.3%), GRU (88.1%), and Bi-LSTM (88.9%), which often struggle with long-term dependencies and lack interpretability. Transformer-based architectures like Vanilla Transformer (89.5%) show slight improvements but remain limited by their computational complexity and lack of adaptive optimization strategies.

Hybrid deep learning models that integrate convolutional layers, such as CNN-LSTM (90.2%) and CNN-GRU (89.8%), demonstrate better feature extraction capabilities, yet they fail to fully optimize sequential dependencies. Attention-based models like Attention-LSTM (91.3%) improve performance by refining temporal dependencies but still fall short compared to our proposed approach.

Optimization-enhanced hybrid models, including LSTM-PSO (91.8%), GRU-GA (91.5%), and Transformer-ACO (92.4%), achieve significant improvements by leveraging heuristic-based optimization techniques for weight tuning. However, these methods can be sensitive to hyperparameter settings and require extensive tuning.

Our proposed model, surpasses all competing models in terms of accuracy, robustness, and explainability. The reduced RMSE (0.072) further indicates improved predictive stability, minimizing error propagation. This confirms the model's ability to handle complex, multimodal, and dynamic data while maintaining high performance and transparency, making it particularly suitable for real-world, high-stakes applications.

6. DISCUSSION

The related work section analyses some of the recent algorithms. These studies attained high performance, including drawbacks like scalability across different data domains and also some studies have faster training times but compromised accuracy. The proposed method have better performance of accuracy with faster training.

Later on in related study stability in training is achieved but with manual tuning of hyperparameters. Later on combining transformers with GANs for improved feature extraction is proposed. This work showed promising results on financial datasets but failed to generalize well across different domains. A scalable framework using distributed computing and federated learning improved efficiency but struggled with accuracy across diverse data sources. A novel self-attention mechanism excelled in trend prediction for structured data but faced challenges with multimodal inputs and real-time updates. Lightweight models designed for edge deployment enhanced efficiency but resulted in reduced accuracy compared to larger models. The proposed method addressed this challenge by introducing the novel guided remora optimization algorithm to avoid manual tunning at same time with better accuracy and also achieving generalization across different domains.

Advancements in trend prediction improved efficiency, accuracy, and adaptability but faced trade-offs in computational cost. Hybrid and cross-domain approaches showed promise but required significant fine-tuning and resources. While our proposed system is able balance well between high accuracy, less time need, low Error rate, automatic weight updation and also achieving generalization across different domains.

Although the proposed framework demonstrates strong performance, several limitations remain. High computational requirements for transformer models could hinder large-scale deployment. Additionally, performance on low-resource domains and less-structured data types may require further refinement. The current interpretability methods, while advanced, may not yet be fully accessible to non-technical users. These limitations guide our future enhancements.

7. CONCLUSION

This study proposed a Generative AI-driven framework for dynamic trend prediction that integrates transformer-based architectures, adaptive learning mechanisms, and the Proposed Guided Remora Optimization Algorithm (PGROA). In line with our core research objectives, the framework was designed to address critical challenges observed in existing models including overfitting, limited interpretability, and inadequate handling of multimodal and cross-domain data.

Experimental validation strongly supports the effectiveness of the proposed framework. The model achieved a high accuracy of 94.8%, an F1-score of 93.8%, and an RMSE of just 0.072, outperforming several state-of-the-art deep learning and hybrid models. These outcomes directly align with our goal to improve predictive precision and robustness across dynamic and heterogeneous datasets. The use of incremental learning and domain-adversarial techniques demonstrated superior adaptability, fulfilling our objective of real-time, cross-domain trend generalization.

Moreover, the incorporation of interpretability methods such as SHAP, attention visualization, and counterfactual analysis addressed the often-overlooked need for transparency in AI-driven forecasting, making the results more understandable and actionable for stakeholders. This aligns with our objective of not just prediction, but *explainable* prediction.

The PGROA component played a pivotal role in dynamically optimizing model parameters with reduced computational cost—demonstrating our contribution toward efficient, scalable learning in contrast to manually tuned traditional models.

Despite these achievements, limitations remain. The computational complexity of transformer-based models still poses a challenge for real-time deployment on edge devices, and further improvement is needed for performance consistency in low-resource or noisy domains. Additionally, while interpretability was enhanced, simplifying explanation outputs for non-technical users remains an open research problem.

Future work will extend the framework by integrating reinforcement learning for continuous policy updates, incorporating broader multimodal datasets, and streamlining inference speed for large-scale, real-time applications. By addressing both performance and practical usability, this research provides a significant step toward a transparent, scalable, and adaptive AI solution for dynamic trend prediction in complex environments.

Compared to incremental advancements seen in recent studies—such as slight tuning of existing models—this framework introduces a fundamentally new combination of adaptive optimization, multi-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

modal generalization, and interpretability. This contributes a deeper, systemic advancement toward real-time, explainable AI systems for trend prediction.

Future directions include developing lightweight transformer architectures for resource-constrained deployment, improving interpretability for non-experts, and addressing adversarial robustness. There is also a need for enhanced domain adaptation techniques to further reduce fine-tuning requirements for new data environments.

Author Contributions

- **Vinodkumar Reddy Surasani**: Conceptualization, Methodology, Software Development, Experimental Design
- **Sarvani Anandarao**: Literature Review, Data Curation, Writing – Original Draft
- **Nagaraju Devarakonda**: Supervision, Validation, Writing – Review & Editing

REFERENCES

[1]. Chandra, Rohitash, Shaurya Goyal, and Rishabh Gupta. "Evaluation of deep learning models for multi-step ahead time series prediction." *Ieee Access* 9 (2021): 83105-83123.

[2]. Hollis, Thomas, Antoine Viscardi, and Seung Eun Yi. "A comparison of LSTMs and attention mechanisms for forecasting financial time series." *arXiv preprint arXiv:1812.07699* (2018).

[3]. Li, Ting, et al. "Autost: Efficient neural architecture search for spatio-temporal prediction." *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020.

[4]. Liu, Lei, et al. "An efficient gan-based multi-classification approach for financial time series volatility trend prediction." *International Journal of Computational Intelligence Systems* 16.1 (2023): 40.

[5]. Smith, Kaleb E., and Anthony O. Smith. "Conditional GAN for timeseries generation." *arXiv preprint arXiv:2006.16477* (2020).

[6]. Shu, Hailong, et al. "Enhancing precipitation prediction accuracy with Transformer-GAN hybrid models." *Fourth International Conference on Machine Learning and Computer Application (ICMLCA 2023)*. Vol. 13176. SPIE, 2024.

[7]. Zhao, Mengyi, et al. "Bidirectional transformer gan for long-term human motion prediction." *ACM Transactions on Multimedia Computing, Communications and Applications* 19.5 (2023): 1-19.

[8]. Liu, Haipeng, et al. "A Trend Detection-Based Auto-Scaling Method for Containers in High-Concurrency Scenarios." *IEEE Access* (2024).

[9]. Alcazar, Javier, et al. "Enhancing combinatorial optimization with classical and quantum generative models." *Nature Communications* 15.1 (2024): 2761.

[10]. Zhao, Liang, et al. "Multi-task learning for spatio-temporal event forecasting." *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2015.

[11]. Zhang, Kexin, et al. "Self-supervised learning for time series analysis: Taxonomy, progress, and prospects." *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[12]. Fahim, Asmaa, et al. "Hybrid LSTM Self-Attention Mechanism Model for Forecasting the Reform of Scientific Research in Morocco." *Computational Intelligence and Neuroscience* 2021.1 (2021): 6689204.

[13]. Yang, Liu, Constantinos Daskalakis, and George E. Karniadakis. "Generative ensemble regression: Learning particle dynamics from observations of ensembles with physics-informed deep generative models." *SIAM Journal on Scientific Computing* 44.1 (2022): B80-B99.

[14]. Chen, Jieyu, et al. "Generative machine learning methods for multivariate ensemble postprocessing." *The Annals of Applied Statistics* 18.1 (2024): 159-183.

[15]. Yao, Zhenjie, et al. "Interpretable trend analysis neural networks for longitudinal data analysis." *ACM Transactions on Computing for Healthcare* 5.2 (2024): 1-13.

[16]. Wikle, Christopher K., and Andrew Zammit-Mangion. "Statistical deep learning for spatial and spatio-temporal data." *arXiv preprint arXiv:2206.02218* (2022).

[17]. Alsharef, Ahmad, et al. "Review of ML and AutoML solutions to forecast time-series data." *Archives of Computational Methods in Engineering* 29.7 (2022): 5297-5311.

[18]. George, A. Shaji, AS Hovan George, and T. Baskar. "Edge Computing and the Future of Cloud Computing: A Survey of Industry Perspectives and Predictions." *Partners*

Universal International Research Journal 2.2 (2023): 19-44.

[19]. Jia, Zidi, Lei Ren, and Yang Tang. "A Contrastive Representation Domain Adaptation Method for Industrial Time-Series Cross-Domain Prediction." *IEEE Transactions on Industrial Informatics* (2025).

[20]. Yazdani, Danial, et al. "Robust optimization over time: a critical review." *IEEE Transactions on Evolutionary Computation* (2023).

[21]. Nott, David J., Christopher Drovandi, and David T. Frazier. "Bayesian inference for misspecified generative models." *Annual Review of Statistics and Its Application* 11 (2023).

[22]. Jin, Ruidong, Xin Liu, and Tsuyoshi Murata. "Predicting popularity trend in social media networks with multi-layer temporal graph neural networks." *Complex & Intelligent Systems* (2024): 1-17.

[23]. Fu, Yuwei, Di Wu, and Benoit Boulet. "Reinforcement learning based dynamic model combination for time series forecasting." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. No. 6. 2022.

[24]. Ji, Xinyao, et al. "The trend-in-trend research design for causal inference." *Epidemiology* 28.4 (2017): 529-536.

[25]. Iwata, Tomoharu, and Atsutoshi Kumagai. "Few-shot learning for time-series forecasting." *arXiv preprint arXiv:2009.14379* (2020).

[26]. Gu, Qitao, et al. "Integrating multi-source transfer learning, active learning and metric learning paradigms for time series prediction." *Applied Soft Computing* 109 (2021): 107583.

[27]. Pöppelbaum, Johannes, Gavneet Singh Chadha, and Andreas Schwung. "Contrastive learning based self-supervised time-series analysis." *Applied Soft Computing* 117 (2022): 108397.

[28]. Zhao, Lifan, Shuming Kong, and Yanyan Shen. "Doubleadapt: A meta-learning approach to incremental learning for stock trend forecasting." *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2023.

[29]. Shen, Yancen, et al. "Uncertainty Quantification Method for Trend Prediction of Oil Well Time Series Data Based on SDMI Loss Function." *Processes* 12.12 (2024): 2642.

[30]. He, Huizi, et al. "A novel crude oil price trend prediction method: Machine learning classification algorithm based on multi-modal data features." *Energy* 244 (2022): 122706.