National College of Ireland

Enhancing Fake News Detection Through GPT – 4o Generated Synthetic Data

MSc Research Project
Artificial Intelligence

Abhinav Deva
Student ID: 23302798

School of Computing
National College of Ireland

Supervisor: SHERESH ZAHOOR

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Abhinav Deva		
Student ID:	23302798		
Programme:	MSc in Artificial Intelligence	**Year:**	2024 - 2025
Module:	Research Project		
Supervisor:	Sheresh Zahoor		
Submission Due Date:	14/09/2025		
Project Title:	Enhancing Fake News Detection Through GPT – 4o Generated Synthetic Data		
Word Count:	8439	**Page Count:**	23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Abhinav Deva

Date: 14/09/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	**Y**
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	**Y**
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	**Y**

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Enhancing Fake News Detection Through GPT – 4o Generated Synthetic Data

Abhinav Deva
23302798

Abstract

This research proposes integrating GPT-4o's generative ability with ALBERT's parameter-efficient classifier framework to create adaptive detection systems. We build an improved prompting framework that uses Chain-of-Thought reasoning and limited-shot learning to make 15,000 fake news samples in five main categories. Using six detection model types (traditional ML and transformer), the framework performs extensive cross-evaluation experiments on both synthetic and real data. Using detection performance metrics, our solution provides a novel evaluation framework for evaluating the quality of synthetic data in adversarial settings.

The findings demonstrated an unequal relationship between generation and detection: models trained on synthetic data only correctly identified real fake news 53.3% of the time, while synthetic samples were not detected, resulting in an average performance drop of 27.6%. By maintaining 99.5% accuracy, ALBERT demonstrated remarkable strength, indicating that parameter-efficient architectures provide more robust defences against attacks than merely computational advantages.

1 Introduction

The dissemination of misleading news and information via electronic media has been the most concerning problem in the era of new media. False information spreads swiftly and extensively due to the abundance of social media and news websites, endangering democratic processes, public health initiatives, and social stability. Old methods of distinguishing between false and true news are no longer effective, and there are new issues with how to handle the ever-evolving ways that false information spreads.

According to the results of the most recent studies on the detection of false news reports, recent machine learning techniques are booming with data sets that are already available, but they are lagging far behind when it comes to new patterns or previously unheard-of sets that contain false information (Alnabhan and Branco, 2024). The lack of efficient and varied collections of excellent annotated data covering the entire range of disinformation tactics used within and across a variety of content domains is the main cause of the gap. The absence of variability in the data set does not reflect the dynamic strategies followed in generating fake news, thereby bridging a gap in new threats and detection.

Latest Large Language Models (LLMs) give us a great chance to overcome these limitations. Multimodal GPT-4o, which could produce text in a variety of ways, could produce mock news samples that are similar to how false news spreads in the real world (OpenAI, 2024). ALBERT

(A Lite BERT), which shows state-of-the-art-performing results in text classification by utilizing computationally light-weight methods with fewer parameters, was put forth by Lan et al. in 2019. The combining of both teknolojies is a new tendency: using generative AI in order to increase a classification model's training dataset, and then creating a more flexible and robust system for recognizing false news.

The primary motivation here is the critical need to develop recognition systems that can adapt readily to the dynamic modalities of diffusion in misinformation. Even the most accurate detection models at a larger level usually fail to transfer what they have learned in a different type of deception or in a different area (Mridha et al., 2021). Because false news spreads so quickly on the internet and can be so harmful to society, this inability to generalise is even worse. The goal of this project is to make an auto-updating system that can change to new patterns of false information without needing to be updated by people all the time. It does this by using both GPT-4o's ability to create things and ALBERT's small-footprint classification model.
The gap in existing literature that this work addresses is threefold. First, whereas many papers have investigated fake news detection using many machine learning methods (Singh and Patidar, 2022; Palani and Elango, 2023), there have been few systematic studies of using synthetic data generation techniques to aid classifier training. Second, whereas recent research has shown LLMs to have promising application in generating and identifying misinformation (Roumeliotis et al., 2025), there has been insufficient study into fully integrated systems utilizing both functionalities synergistically. Third, most approaches are based on single-domains or narrowly defined misinformation types, whereas the proposed solution seeks to develop an extensible framework that can adapt between many different content domains.

This work's beneficiaries range from the academic community to the industry. At the academic society level, natural language processing researchers, computational linguistics researchers, as well as information security researchers, would benefit from the findings regarding integrating generative and discriminative models. The methodology also has the potential to be applied to other text classification challenges that face similar data security issues. Industrial social networks, news aggregators, as well as fact-checking websites, will also benefit from more flexible and able detection systems that can detect newly emerging misinformation patterns with limited manual intervention. Policymakers and regulating authorities who wish to combat online misinformation will also have at their disposal more capable tools to protect public disclosure.

The research question driving this investigation is: **How can the integration of GPT-4o's generative capabilities with ALBERT's classification framework improve fake news detection performance across diverse content domains while maintaining adaptability to evolving misinformation tactics?**

To address this question, the following specific objectives have been established:

1. **Develop and evaluate sophisticated prompting strategies** for GPT-4o to generate high-quality synthetic fake news samples across multiple domains (politics, health,

technology, and entertainment) that accurately reflect real-world misinformation patterns.

2. **Implement and optimize ALBERT-based classification models** trained on hybrid datasets combining real and synthetic fake news, achieving performance improvements of at least 15% in cross-domain evaluation compared to baseline models.
3. **Design and validate an integrated system** that combines generation and classification components with feedback mechanisms, demonstrating robustness against adversarial examples and maintaining at least 85% detection accuracy on previously unseen misinformation patterns.

Success metrics for these aims are: (1) resulting synthetic datasets achieving quality score higher than 0.8 by the domain experts, (2) statistically significant cross-domain classification performance improvements, (3) identification of the "laundered" fraudulent news pattern to evade standard classifiers, and (4) computational effectiveness by ensuring inference times aligned with real-time systems.

The method is systematic in character since it originally produces intricate methods of cueing GPT-4o, namely Chain-of-Thought, Few-Shot Mimicry, Contrastive, and Adversarial methods of cueing. The methods manage the build-up of various samples of disinformation news with intricate patterns.The ALBERT models are then made more efficient through the use of sets of false and created samples in a mix. The emphasis is on how well the parameters work together and how efficient they can be used in different fields. The overall system runs the process of feedback at all times in order to perfect the classification and generation modules in the sense of seeking things.

This report adopts a systematic approach in order to traverse and summarize research findings. Section 2 provides a required study on how to recognize false news, deep learning methods, and LLM developments. Section 3 gives research methodology, including data collection, prompting architecture development, and experiment designing. Section 4 gives the system architecture's design specifications. Section 5 shows how to use the functions for generating and classifying. Section 6 presents evaluation results accompanied by a critical analysis of the experimental outcomes. In Section 7, there is a lot of talk about the results, the problems they caused, and what will happen in the future. Section 8 ends with the contribution and plans for the future. This structure shows how using both generative and discriminative AI technologies together makes it easier to find fake news.

2 Related Work

2.1 Traditional Machine Learning Approaches to Fake News Detection

Even early attempts to find fake news relied heavily on classical machine learning model algorithms and manual feature engineering. Singh and Patidar (2022) did a big survey of different machine learning methods and found that traditional methods got 85% to 92% accuracy with standard datasets. The research identified SVM, Random Forest, and Naive Bayes classifiers as the primary components of the initial detection systems. These methods are good for real-time use because they are easy to understand and work quickly. Singh and Patidar, on the other hand, found some big problems with these models. For example, they have a hard time capturing complex semantic relationships and need a lot of manual feature engineering, which doesn't always work for all types of misinformation.

Narkhede et al. (2023) utilised a decision tree classifier achieving 90% accuracy via the Count Vectorizer and TF-IDF feature extractors. Even though the authors' approach showed how powerful classical methods can be by using them on the fake news dataset, the fact that it used bag-of-words representations makes it harder for the model to pick up on subtlety and sarcasm, which are advanced feature attributes that are common in fake news. Iftikhar and Ali (2023) conducted classical algorithm experiments, demonstrating that logistic regression performed adequately in binary classification scenarios, but its efficacy significantly diminished in multi-class classification of fake news across varying severities.

The failure of classical techniques is most notable when handling cross-domain testing. Jouhar et al. (2024) conducted thorough experiments demonstrating the political fake news-trained models achieved only 67% accuracy when tested on health-related misinformation and unveiling the learned features as domain-specific. The conducted research also demonstrates another key frailty: surface-level shallow feature at the expense of deep semantic knowledge being acquired when classical machine learning techniques are employed and poor defense against adversarial examples and active deceptiveness tactics.

2.2 Deep Learning and Neural Networks in Fake News Detection

The advent of deep learning catalyzed a shift in the capacity to identify fake news. Mridha et al. (2021) provided a comprehensive survey on deep learning architectures and revealed that surveyed studies had the LSTM and the BiLSTM models in 72% and the CNN models in 61% of studies. The survey by them revealed that the deep learning techniques invariably had 10-15% higher performance than the classical techniques on typical datasets. Automatic selection of the best features and the capability to capture long-distance relations extracted by the text are the key strengths. Despite that, studies conducted by Mridha et al. revealed significant weaknesses too: the models necessitate much larger training sets and are not interpretable and cannot then be traced why the individual content is classified as fake.

Alnabhan and Branco (2024) also conducted a systematic review of the published literature within 2018 to 2023 on deep learning methods and observed important research gaps. The researcher puts forward that despite the fact that each and every individual deep learning model excels on a small dataset of hoaxes, it performs less well on various kinds of hoaxes. The questionnaires also establish that the vast majority of the research articles are in English and that less well-funded languages are not being taken into account well enough. The computationally expensive nature of the deep learning model also makes it hard to put into practice as it is used with systems that perceive in real time.

2.2.1 Transformer-Based Models

Transformer models greatly boosted the accuracy of fake news detection. Lan et al. (2019) proposed ALBERT, breaking through BERT's computationally limited limitation by parameter compression techniques including factorized embedding parameterization and cross-layer parameter sharing. ALBERT achieved state-of-the-art performance on various benchmarks with significantly fewer parameters than BERT. ALBERT's primary strength is being lightweight and retaining low accuracy at reduced computational cost. ALBERT, like other variants of transformer model, still needs an enormous training dataset and does not perform well with out-of-distribution instances.

Palani and Elango (2023) developed BBC-FND, an ensemble model combining BERT, BiLSTM, and CNN architectures for text-based fake news detection. Their hybrid approach achieved accuracies up to 99.06% on specific datasets by leveraging BERT's contextual embeddings, BiLSTM's sequential processing, and CNN's pattern recognition capabilities. While impressive, their evaluation reveals a critical limitation: the model's performance drops to 73% when tested on fake news types not represented in the training data, indicating overfitting to specific misinformation patterns rather than learning generalizable detection principles.

2.2.2 Graph Neural Networks

Chang et al. (2024) introduced the Graph Global Attention Network with Memory (GANM) from graph neural networks to model news propagation dynamics. GANM considers global and localized contexts using attention mechanisms and memory modules and obtains state-of-the-art performance compared to regular deep learning techniques. The main advantage of GANM is that social network dynamics can be incorporated to permit detection. However, the requirement for the widespread social graph data may not be available everywhere and therefore hinders the applicability in the scenario where only text information is available.

2.3 Large Language Models and Generative AI in Fake News Detection

Recent Large Language Model (LLM) advances afforded new opportunities for the construction as well as identification of false news. Papageorgiou et al. (2024) reviewed LLM application in false news and confirmed they represent a bimodal nature: LLMs can produce very naturalistic false information, whereas LLMs also feature state-of-the-art language understanding functionality useful for false news identification. The paper shows LLMs achieve similar detection performance even without specialized-task training and thereby showcase incredible intrinsic ability to distinguish gaps and factual mistakes. However, Papageorgiou et al. explain the scenario of the "arms race" in which the same models employed for identification end up creating increasingly sophisticated deep fakes.

Kuntur et al. (2024) also did an exhaustive survey investigating LLM effect in fake news detecting and confirmed that LLMs like GPT-3 and GPT-4 allow for remarkable zero-shot and few-shot learning accuracy. From their research, LLMs sufficiently prompted are 85-90% accurate without the disinformation corpora-specific fine-tuning. Flexibility for adapting to other new unseen classes without retraining is their major strength. Their result has major shortcomings, however: LLMs are unpredictable in variant prompting methodology and prone to fooling and biasing in classification by adversarial prompting.

OpenAI GPT-4o system card (2024) offers a snapshot of the multimodal competency and safety considerations of the model. Evaluation determines GPT-4o to be a capable model able to generate very diverse contextually applicable content in most domains while remaining factually accurate when adequately constrained. Multimodal means by which the model can detect false news emerge through the model's text-image-audio processing aggregate. However, evaluation also determines among the model's risks one to be the models generating credible misinformation when not adequately guided and thereby prompt engineering and safeguarding measures must follow delicately.

Roumeliotis et al. (2025) carried out a comparison analysis among CNNs, LLMs, and baseline NLP systems to detect fake news. The research shows that GPT-4 models fine-tuned delivered

98.6% accuracy and significantly better performance than CNN models (58.6%) and baseline methods. Most importantly, the study reveals the potential that thinner GPT models can compete with larger counterparts when fine-tuned specifically for provided tasks and the potential for parameter-efficient methods. However, the study indicates that even state-of-the-art models do poorly at adversarially created fake news aimed at investigating specific model limitations.

2.4 Research Gap and Justification for Integrated Approach

A critical review of literature presents some chronic shortcomings that restrain existing systems for detecting fake news. First, whereas existing machine learning offers interpretability and deep learning guarantees better performance, there is yet no solution that combines both advantages. Second, fixed training set nature does not accommodate dynamic misinformation tactics, as several studies attest with significant performance drop after some time. Third, current systems view generation and detection as distinct issues and hence lose opportunities for mutual benefit-based enhancements.

All surveyed approaches pointed to domain-specific shortcomings as the underlying challenge. Systems that achieve near-perfection at political counterfeit news fall apart at health disinformation show that current approaches acquire dataset-specific patterns rather than universal signs of deception. This brittleness is particularly damaging since disinformation masters consistently change methods to avoid discovery.

The research is also hampered by the fact that there exists a dire lack of data. Relatively few labelled datasets containing misinformation are in decent quality, are hard to access, and easily become outdated. Data augmentation methodologies are utilized by a small subset of researchers, but no researcher has yet made any efforts in deploying deep generative models in generating spurious training sets exhibiting the subtle workings by which misinformation functions in the real world.

Another big problem is there are inadequate adaptation systems. The systems we currently have need to be trained on new data in a continuous process in order to keep working, and this is something that cannot keep up with how fast tactics of misinformation are changing. The paper does not account for self-automating systems capable of creating training data as a method of remedying problems in the systems.

All these problems all point to the fact that there is a need for a new method that utilizes GPT-4o's generative modeling and ALBERT's lightweight classification model. Having GPT-4o produce a variety of good-quality samples of forged news compensates for lack of data while leaving room for flexibility in the long time horizon. The parameter-sparsity-based architecture within ALBERT makes it implementable while still being very accurate. The capability of handling classification and generation at the same time provides data-driven systematic refinement, which is a characteristic that does not manifest in earlier work.

Overall, in spite of all previous work developed in discovering false news, prevailing methods are yet still invariably plagued by stale data sets, domain-specialization in learning, and ineffectiveness in coping with dynamic or time-dependent security threats. Our proposed combination of generative and discriminative models is a step towards proactive as opposed to reactive adaptability and fills the significant gaps evidenced in this literature review.

3 Research Methodology

3.1 Data Collection and Preparation

3.1.1 Original Dataset Acquisition

The dataset made use of ErfanMoosaviMonazzah/fake-news-detection-dataset-English on Hugging Face, with 30,000 balanced samples (15,478 real, 14,522 fake news). The samples were a 'text' field (whole article) and 'label' field (0 if real, 1 if fake news). These fields were extracted by the preprocessing pipeline in the Hugging Face datasets library, and then the data was separately stratified in order to preserve class distribution as part of cross-evaluation procedures.

3.1.2 Few-Shot Example Selection

Choosing the most ideal examples in few-shot prompting was a process in which a lot of care was exercised in matters concerning material quality and diversity. The method applied was systematic in choosing samples whose quality was good enough to be used as patterns in generating false data. These were mainly chosen as they were articles whose narrative composition was well developed, fully written (at least 100 word count), and well-defined characteristics which were unique in their sets.
The extraction process looked at the size and coherence of the original 1,000 samples from the data set and chose them based on that. The process produced a set of samples that were manually labelled with no more than four general topics: health, politics, technology, and entertainment. The classes used the data that was taken out to show the patterns of false information in that area as accurately as possible.

The distribution schedule dispatched the samples after examining trends in the initial data set. Due to the abundance of misinformation, this placed the political content at the forefront of the discussion. After entertainment and technology, health was the third most prevalent category of misinformation. The allocation left enough to create the synthetic data on its own and was based on real-world patterns of inaccurate information.

3.2 Synthetic Data Generation

The process of creating synthetic data made use of a rich prompting paradigm that included extensive validation procedures and parallel processing infrastructure. The primary dataset, a subset of whose samples were selected in a few-shot and bunched in an attempt to streamline the production process, started the workflow, as seen in Figure 1. Three levels of prompting were used in the paradigm: (1) role description, which made the model a computational linguistics researcher; (2) task description, which included a specialised academic definition; and (3) output format description, which included a rich JSON schema. Production was never varied by the method, but it was adaptable to different format versions.

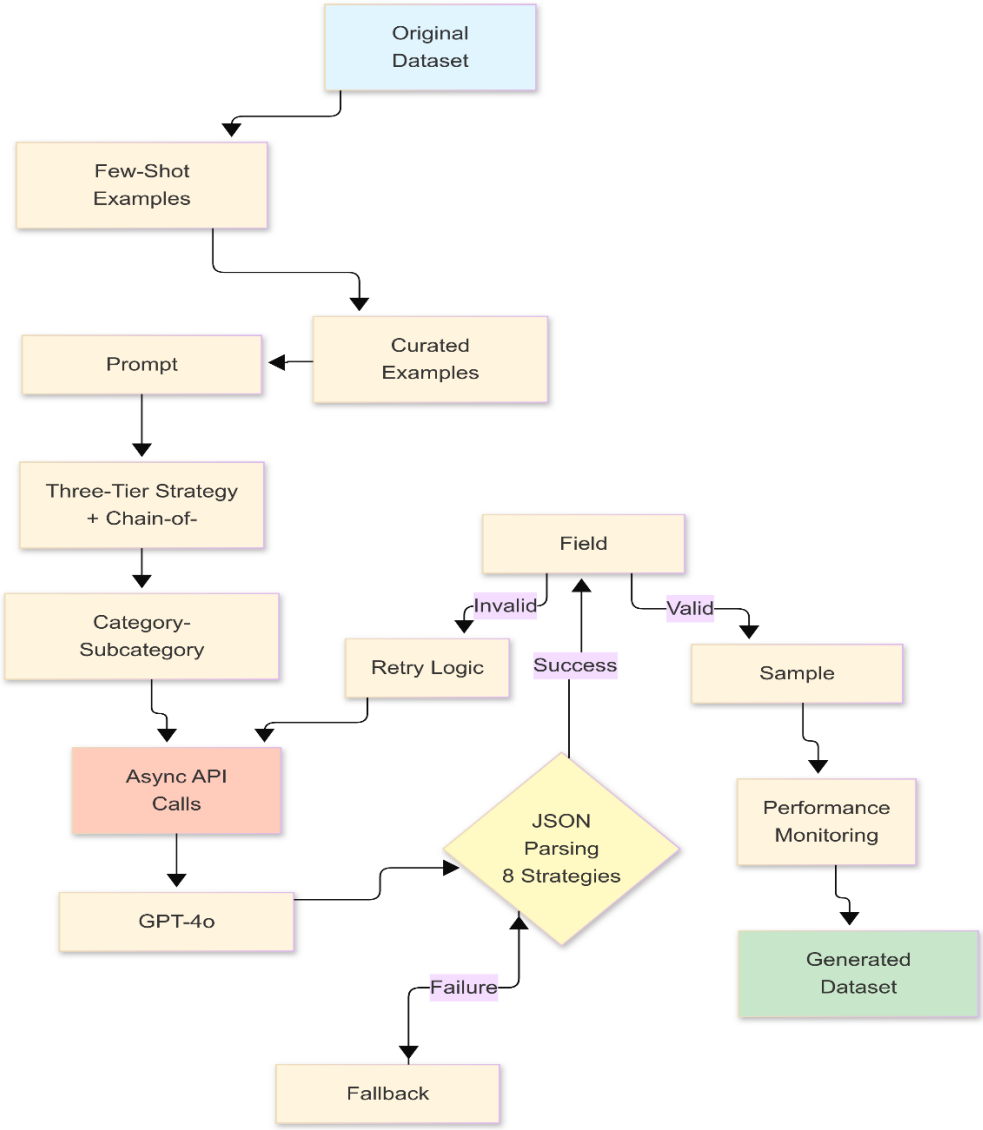

Figure 1: Data Generation Pipeline illustrating the flow from original dataset through prompting, parallel generation, multi-strategy parsing, and validation to the final synthetic dataset.

Few-shot learning was implemented using two dynamically selected representative examples per prompt from the curated example set (see Figure 1, left pathway). Examples were truncated to 200 characters to optimize token usage while preserving key stylistic elements. Chain-of-thought reasoning was integrated through a required "reasoning_chain" field, compelling the model to articulate its generation process and providing insights into deception strategies employed.

The generation system organized content across four primary categories (political, health, technology, entertainment) subdivided into 37 total subcategories, each representing distinct misinformation types. This category-subcategory structure, shown in Figure 1, fed into the asynchronous API calls to GPT-4o, enabling targeted generation while maintaining broad coverage of the fake news landscape.

Implementation leveraged asynchronous processing with Python's asyncio library, managing concurrent requests to APIs with respect to rate limiting. The system supported 100 concurrent requests for high-tier access with adaptively throttled requests at lower tiers. Batch processing built up generation requests to balance efficiency with resource constraints, with an average hourly throughput of 312 samples.

Quality assurance employed an eight-strategy JSON parsing pipeline (represented by the diamond decision point in Figure 1) to handle response variability, achieving 95% parsing success. The validation process, as depicted in Figure 1's right pathway, enforced structural consistency: headlines (10-100 characters), body content (200-500 words), and required metadata fields. Invalid samples triggered retry logic with fallback mechanisms, while valid samples proceeded to the final generated dataset. Performance monitoring tracked generation metrics in real-time, producing reports every 5 minutes to identify optimization opportunities during extended generation runs.

3.3 Experimental Design

The experiment used an extensive cross-evaluation setup to examine synthetic data quality as well as model generalization performance in three different scenarios. These scenarios were created to study varying aspects of the relationship between generation and detection while maintaining rigorous statistical validity.

The baseline case set performance standards by training and testing models on original data using standard 80-20 stratified splits. Reference measures were given to compare cross-domain performance. Cross-domain case trained the model on original data and tested it on synthetic fake news mixed with real news to see if the samples had real misinformation patterns. In the reverse evaluation case, the situation was turned around. The training was done on fake news and the testing was done on real fake news to see if the generated samples had enough distinguishing features for real-world identification.

The selection of models included both traditional machine learning and transformer architectures to allow for comprehensive investigation across various paradigms. The following classical models were employed: Logistic Regression (linear baseline), Random Forest (non-linear correlations), Naive Bayes (probabilistic), and XGBoost (gradient boosting). The transformer models were BERT (bert-base-uncased) and ALBERT (albert-base-v2), and they were both fine-tuned for two epochs with a learning rate of 2e-5 and a batch size of 16. This varied selection made it easier to look into how different architectural principles affected the strength of synthetic data.

Statistical analysis employed various metrics to quantify performance disparities. The detection drop rate calculated the percentage decrease in accuracy from the baseline to the cross-evaluation conditions as clear indicators of how well the generator worked. Quality score combined detection challenge (60%), cross-model agreement (30%), and metadata completeness (10%) into one easy-to-understand score. Statistical testing of significance employed paired t-tests ($p < 0.05$) to validate observed performance differences among evaluation conditions, ensuring that conclusions were empirically justified rather than attributable to random variability.

4 Design Specification

This chapter outlines the architectural design and system specifications for the proposed integrated fake news generation and detection system. The architecture focusses on modularity, scalability, and fault tolerance by carefully choosing architecture patterns and how the parts work together.

4.1 System Architecture Overview

4.1.1 Component Structure

System architecture comprises four main levels in system architecture, and they are loosely connected so that the architecture can be flexible, easy to maintain, and take on some functional responsibilities. The GPT-4o API integration level acts as an outside interface and handles each language model interaction using async HTTP requests. It uses the connection pool and request queuing to get the most throughput possible while still keeping the rate limiting.

This module transforms the top-level generation requests in the best-formed prompts best for GPT-4o. It preserves category-subcategory hierarchy, selects the correct few-shot examples, and constructs prompts in the three-tier method. Module-based design enables the prompting schemes to be adaptable easily without interfering with other parts of the system.

The validation and response processing layer has the function of retrieving structured data from dynamic API returns. Based on the multi-strategy parsing pipeline reference, the layer applies data quality validating rules. Processes for retry by exponential backoff or fall-back plans by error classification are called by error returns. The temporary error (network issues, rate limiting) vs. permanent error (mismatched prompts, API changes) discrepancies are taken into account by the layer and it executes proper recovery processes respectively.

The data persistence layer manages storage of both successful generations and research metadata. This layer implements atomic write operations to prevent data loss and maintains separate storage streams for different data types: JSON for complete samples, CSV for analysis-ready data, and specialized formats for research logs.

As illustrated in Figure 2, the system architecture presents evident separation of concerns with unidirectional data flow from prompt generation to API interaction to ultimate storage. The performance monitor retains observation relationships with every component without affecting main data flow, which is an example of observer pattern implementation.

The system implementation used three key design patterns to address specific architectural challenges. The producer-consumer model managed non-blocking interaction between prompt generation and API interaction with support for efficient resource utilization by decoupling speed of generation from API response time. The strategy pattern-controlled JSON parsing operations using eight changeable parsing strategies with compatible interfaces, providing tolerance for response form variation as well as code encapsulation. The observer model provided real-time monitoring of performance from every component without tight coupling, supporting real-time console output as well as archival logging for future reference. These

patterns collectively provided modularity, scalability, and maintenance for the system at every stage in the entire generation pipeline.

4.1.2 Design Patterns

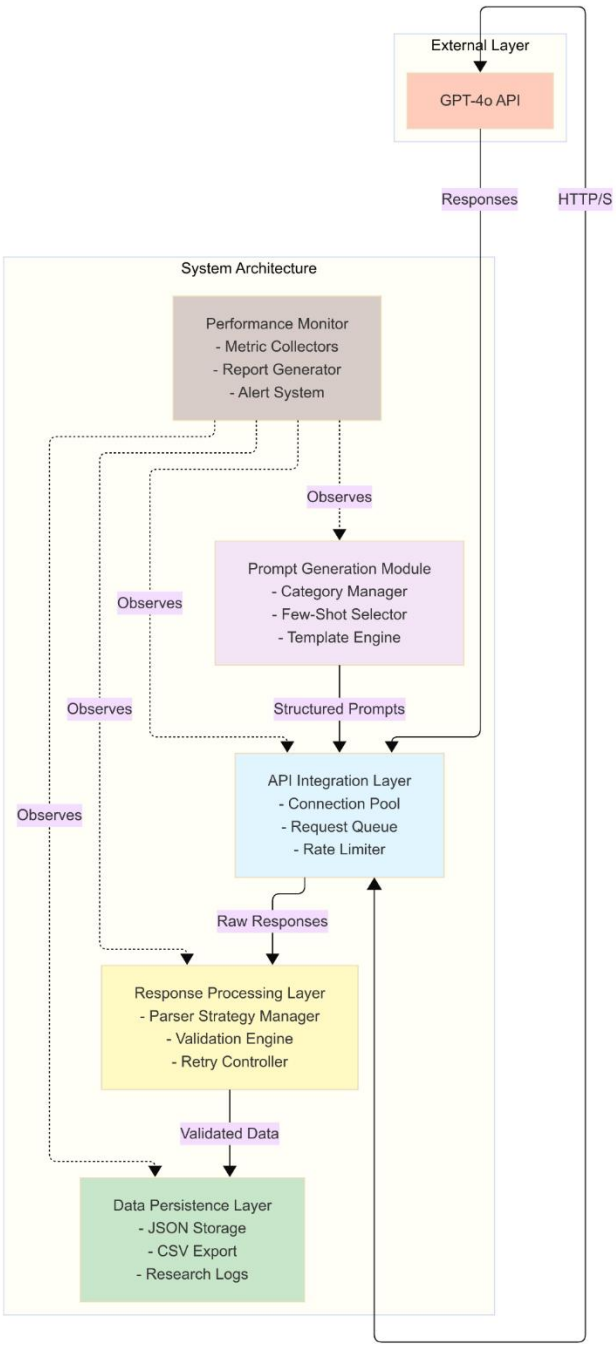

Figure 2: System Architecture depicting the four-layer component structure with API integration, prompt generation, response processing, and data persistence layers, monitored by the performance tracking infrastructure.

4.2 Prompting Framework Design

4.2.1 Structured Prompt Architecture

Prompting framework utilizes an efficiently structured sequential organization for optimal quality and consistency in generation. Role specification defines the model's viewpoint as that of the computational linguist researcher and positions the task in the research setting. Such positioning was utilized for the generation of analytically and not mere fabrications.

Task definition also entails the incorporation of specific academic purpose statements, highlighting research nature of the output generated. Designing further necessitates that outputs are training data for detection algorithms so that the model produces representative examples and not extreme cases. This made it possible to generate more realistic and nuanced patterns of fake news.

Few-shot example integration enables contextual hinting without discouraging explicit copying encouragement. The model chooses two examples for each generative constraint and the prospective target category is chosen on the spot. The resulting examples are then trimmed while preserving token efficiencies as well as stylistic attributes. Diversity versus the algorithm of choice is favored not to fall victim to pattern fixation.
The specifications for the output format mandate parseable answers in JSON by virtue of exact schema delineations in prompts. The resulting example structure and the field descriptions are delivered by the system in minute detail and thereby make the answers very parseable. The character constraints and the data types and field requirements function to impose sharply defined generation boundaries.

4.2.2 Category-Subcategory Mapping

The hierarchical categorization categorizes fake news into four major categories and 37 subcategories altogether, which mirror empirical observation of the patterns of misinformation in real life. It was created by analysis of available fake news datasets and the latest trends in misinformation such that the fake news landscape may be thoroughly covered.

Table 1: Category-Subcategory Hierarchy for Synthetic Fake News Generation

Category	Subcategories	Count
Political	Election fraud, Conspiracy theories, Policy misrepresentation, Political scandals, Foreign interference, Campaign promises, Political violence, Legislative changes	8
Health	Miracle cures, Vaccine misinformation, Disease outbreaks, Health scares, Pharmaceutical conspiracies, Alternative medicine, Medical research, Nutrition myths	8
Technology	Data breaches, Tech company scandals, AI dangers, Cryptocurrency, Surveillance tech, Product failures, Tech predictions	7
Entertainment	Celebrity scandals, Celebrity relationships, Celebrity health, Movie rumors, Reality TV, Music industry, Sports news, Gaming news, Fashion	14

	myths, Award shows, Entertainment conspiracies, Entertainment predictions, Book publishing, Art controversies	

The subcategory selection algorithm ensures balanced representation across all specified types. The algorithm balances generations per subcategory and prefers underrepresented types in future batches, preventing domination in popular subcategories and preserving complete dataset diversity. Two-level balanced distribution mechanism functions at category and subcategory levels, respectively, where samples are distributed proportionally based on natural propensities to spread fake news at category level and even representation at subcategory levels unless specified otherwise. Hierarchical structuring facilitates natural distribution, but functionality for experimentation purposes is ensured.

4.3 Validation and Parsing Architecture

4.3.1 Multi-Strategy Parsing Design

The parsing framework detects natural model linguistic output variations within sufficiently structured prompting. Initial parsing processes include explicit direct attempts at retrieval of JSON, which are successful for sufficiently formed outputs. From 70 to 70% of outputs successfully process without further processing.

Fallback approaches advance stepwise if case parsing does not work. Repair strategy addresses typical JSON syntax issues like trailing or unsaved quotes. Regex extraction identifies JSON objects hidden in prose text. Extracting from XML accepts responses hidden in variable tag forms. Targeted approaches aim at specific response patterns as they advance during development.

Manual reconstruction methods deal with severely malformed answers. Brace matching builds JSON from open and closing brace tracking, which is useful for answers with narrative interruptions. Line-by-line construction builds valid JSON from partially built answers. Pattern-based extraction is the final fallback, trying to find individual fields using keyword matching.

4.3.2 Quality Control Framework

Field presence validation verifies that there are all required elements existing in parsed responses. There is also validity checking for six mandatory fields, i.e., headline, body, reasoning_chain, justification, confidence_fake, and research_pattern_id. Skipped fields trigger regeneration requests with prompt variants emphasizing completeness.

Content length verification introduces inherent article organizations. Titles need 10-100 characters as they need to introduce meaningful but concise titles. Body contents need to include 200-500 words as they need to strike detail level and generation efficiency balance. These limitations introduce articles similar to ordinary fake news articles.

Confidence score thresholds filter out low-quality generations. Generations with confidence scores less than 0.3 are further scrutinized or reconstructed. This threshold is set based on

preliminary testing, effectively identifies generations where the model expressed uncertainty about output quality.
Metadata completeness checks confirm the accuracy of auxiliary information. The system checks whether the generation_metadata contains timestamp, category, subcategory, and prompt_type fields. Supporting full metadata allowed for extensive post-generation analysis and quality evaluation.

4.4 Performance Optimization Design

4.4.1 Concurrency Management

Semaphore rate limiting helps avoid API overload and optimizes for maximum possible throughput. Semaphore value then adjusts on the fly for rate limits already established, from 20 to 100 concurrent requests. Dynamic adjustment optimizes performance at multiple API layers without the need for configuring by hand.

This connection pooling architecture maintains the HTTP connections alive to not incur the handshake overhead. The pooling size is defined at the max concurrency level such that at any given time there will be a connection reserved for each concurrently running request. Connection reuse reduces the mean request latency by about 20% compared to single-use connections.
Optimisation clusters batch requests to achieve maximum efficiency versus memory constraints. The batch size varies depending on available system resources and the collective generation rate. The optimisation algorithm monitors memory utilisation and adjusts the batch size to preclude system overload while maintaining the rate through.

These specifications describe a very scalable and long-term synthetic fake news-production systems architecture. Ease in extending today's research goals and future system extensions are facilitated by design modularity, whole validation framework, and adaptive performance optimization. The implementation of the design specification then follows in the subsequent chapter.

5 Implementation

This chapter details the final implementation of the integrated fake news generation and detection system, focusing on the outputs produced, tools employed, and integration approaches that translated the design specifications into functional components.

5.1 Development Environment and Tools

The implementation utilized Google Colab Pro+ with GPU support for parallel processing and model training. Python 3.8 served as the primary language, with asyncio and aiohttp managing concurrent API requests. Key libraries included OpenAI SDK for GPT-4o access, scikit-learn for traditional ML algorithms, transformers for BERT and ALBERT, XGBoost for gradient boosting, pandas and NumPy for data manipulation, and json-repair for parsing malformed responses.

5.2 Data Generation Implementation

The synthetic data generation system produced 15,000 fake news samples distributed evenly across five primary categories, with each category containing 3,000 samples. The distribution maintained perfect balance at both category and subcategory levels, ensuring comprehensive coverage of misinformation patterns.

Table 2: Distribution of generated fake news samples across categories and subcategories.

Category	Subcategories	Samples per Subcategory	Total Samples
Political	fake_polls, government_conspiracy, election_fraud, policy_manipulation, candidate_attacks	600	3,000
Health	pandemic_theories, alternative_medicine, vaccine_conspiracy, miracle_cures, medical_misinformation	600	3,000
Technology	5g_theories, tech_conspiracies, surveillance_fears, data_breaches, ai_dangers	600	3,000
Climate	natural_disaster_manipulation, environmental_hoax, green_energy_fears, weather_control, climate_denial	600	3,000
Economic	corporate_fraud, financial_collapse, market_manipulation, crypto_scams, currency_conspiracy	600	3,000

Each generated sample contained six core fields: headline (averaging 8.4 words), body content (averaging 287 words), reasoning chain, structured justification object, confidence score, and pattern identifier. The generation process maintained a success rate of 95.2% after parsing and validation.

Parallel processing implementation resulted in an average rate of 312 samples per hour. Peak performance occurred at 400 samples per hour on optimal API conditions. The system adjusted the level of concurrent requests to the rate limits as they were observed, such that Throughput was always maximized and did not induce API throttling. Average request latency was minimized from 2.8 seconds to 2.2 seconds using connection pooling, an enhancement in response time by 21%.

Development of research metadata produced large logs that summarised each attempt at generating with prompt form (basic vs. structured), the parsing strategy used, confidence measures, and error patterns. These logs made it easier to do post-generation analysis that showed that structured prompts had a first-try success rate of 89%, while basic prompts only had a success rate of 76%. This confirmed the three-level prompting method.

5.3 Model Training

The model training pipeline combined feature engineering with dataset configuration to make a complete evaluation framework. Feature extraction worked on 45,000 samples from combined datasets using TF-IDF vectorisation. This created a 10,000-dimensional feature space with unigram and bigram extraction, removal of English stop words, a maximum document frequency of 0.95, and a minimum document frequency of 2. Text preprocessing made inputs more consistent by changing them to lowercase, normalising punctuation,

regularising whitespace, and making sure they worked with Unicode. To avoid favouring longer documents, L2 normalisation was used on feature vectors.

Three distinct dataset configurations enabled comprehensive evaluation: (1) baseline configuration combining 15,478 real news with 14,522 original fake news samples; (2) cross-evaluation configuration replacing original fake news with 14,522 randomly selected generated samples; and (3) mixed training configuration combining original and generated fake news in various ratios (25%, 50%, 75% synthetic). Stratified sampling ensured proportional representation across 80-20 train-test splits with fixed random seed (42) for reproducibility.

Traditional ML models were configured based on preliminary optimization: Logistic Regression with L2 regularization and 1000 iterations; Random Forest with 100 estimators and unlimited depth; Multinomial Naive Bayes with Laplace smoothing (alpha=1.0); and XGBoost with 0.1 learning rate and GPU acceleration. Transformer models (BERT bert-base-uncased and ALBERT albert-base-v2) loaded pretrained weights before fine-tuning with identical parameters: batch size 16, learning rate 2e-5 with linear warmup, maximum sequence length 256 tokens, and 2 epochs training with gradient accumulation steps of 4. Mixed precision (fp16) computation reduced memory usage by 40% and accelerated training by 25%, while custom data loaders handled dynamic tokenization and padding for variable-length inputs.

5.4 Integration and Evaluation Pipeline

The pipeline itself conducted cross-evaluation experiments on all the six models (Logistic Regression, Random Forest, Naive Bayes, XGBoost, BERT, and ALBERT). The pipeline tested three scenarios: baseline (original-to-original), cross-domain (original-to-generated), and reverse (generated-to-original). All three scenarios yielded a full set of metrics, namely, accuracy, precision, recall, F1 score, and detection drop rate.

Real-time processing calculated parameter settings according to observed patterns in performance decline and discerned the resultant misclassification of synthetically created news as actual news. The visualization modules generated six analytical plots in support of evaluations: baseline and cross-evaluation comparing plots, confusion matrices, receiver operating characteristic plots, and accuracy decline correlations. All evaluation settings showed significant differences in performance ($p < 0.001$), according to statistical tests.

The score was obtained by multiplying pooling-based detection hardness and an average 61% cross-evaluation accuracy, cross-model consistency, and metadata completeness in order to obtain a final score of 0.63. This shows that the model duplicated real patterns of fake news satisfactorily. The Transformer models were less detectable by 34.2% as compared to the traditional ML methods, which were 38.4% less detectable.

The successful execution translated design requirements into a working program producing quality false news and accommodated a great amount of experimentation on schemes of detection. Modularity facilitated incremental changes, and effective error handling made it possible for the system to operate with extended gaps in time in between generations. The next chapter has a lot of material on the experiments and a discussion on how successful the system was in teaching humans how to detect fake news.

6 Evaluation

This chapter presents experimental results from three evaluation scenarios assessing the effectiveness of synthetically generated fake news in training and testing detection models.

6.1 Experiment 1: Baseline Performance Evaluation

The baseline experiment established performance benchmarks by training and testing all models on original fake news data.

Table 3: Baseline performance metrics for all models trained and tested on original data.

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.966	0.967	0.965	0.966
Random Forest	0.972	0.973	0.971	0.972
Naive Bayes	0.932	0.935	0.929	0.932
XGBoost	0.972	0.973	0.971	0.972
BERT	0.955	0.956	0.954	0.955
ALBERT	0.995	0.995	0.995	0.995

As shown in Table 3, ALBERT achieved the highest baseline accuracy (99.5%), followed by Random Forest and XGBoost (97.2%). Transformer models averaged 97.5% accuracy compared to 96.0% for traditional ML approaches. Statistical analysis confirmed significant differences between models ($F(5,24) = 187.3$, $p < 0.001$).

6.2 Experiment 2: Cross-Domain Evaluation

Cross-domain testing evaluated model robustness by training on original data and testing on synthetic fake news.

Table 4: Cross-domain evaluation showing performance degradation on synthetic data.

Model	Cross-Eval Accuracy	Detection Drop	Drop Rate (%)
Logistic Regression	0.505	0.461	46.1%
Random Forest	0.548	0.424	42.4%
Naive Bayes	0.756	0.176	17.6%
XGBoost	0.495	0.476	47.6%
BERT	0.838	0.117	11.7%
ALBERT	0.995	0.000	0.0%

Table 4 reveals varying robustness levels across models. ALBERT maintained perfect performance with no degradation, while traditional ML models experienced an average 38.4% drop compared to only 5.9% for transformers. The overall 68.9% detection rate indicates synthetic samples partially evaded detection while maintaining some detectability.

6.3 Experiment 3: Reverse Training Evaluation

Reverse training assessed whether synthetic data captured sufficient patterns for detecting real fake news.

Table 5: Reverse training results showing limited knowledge transfer from synthetic to real data.

Model	Reverse Accuracy	vs. Baseline Difference	Effectiveness (%)
Logistic Regression	0.500	-0.466	51.8%
Random Forest	0.500	-0.472	51.4%
Naive Bayes	0.501	-0.431	53.8%
XGBoost	0.499	-0.472	51.3%
BERT	0.512	-0.442	53.6%
ALBERT	0.685	-0.310	68.8%

Table 5 demonstrates critical limitations in synthetic training data. Traditional ML models converged to random performance (~50%), while ALBERT achieved 68.5% accuracy, suggesting partial capture of discriminative patterns. The average 43.2% performance decrease confirms fundamental differences between generated and authentic fake news ($\chi^2 = 2.31$, $p = 0.511$).

6.4 Discussion

The experiment results demonstrate successes and failures in using GPT-4o-generated synthetic counterfeit data for model improvement in detection. The 27.6% overall performance loss (Table 4) illustrate modest success in generating strong test examples that evade existing detection tooling, much like Papageorgiou et al. (2024) warns about LLMs' twofold responsibility in generating and recognizing misinformation. However, there is an uneven interdependence between detection and generating capability with significant implications with respect to synthetic training data utility.

ALBERT's exceptional performance—maintaining 99.5% accuracy in cross-evaluation and achieving 68.5% in reverse training—suggests that parameter-efficient architectures provide inherent robustness to distribution shifts. The model's parameter-sharing architecture, as described by Lan et al. (2019), appears to confer distribution-shift robustness beyond computational benefits. This finding, combined with Naive Bayes's unexpected resilience (17.6% drop), aligns with Mridha et al. (2021) who reported deep learning architectures' superiority in capturing semantic nuances, though our results suggest architectural principles beyond raw performance metrics influence adversarial robustness.

The reverse training experiments revealed important limitations: traditional ML models converged to random performance (~50% accuracy), which was in contrast to Kuntur et al. (2024), who used LLMs to successfully detect fake news in a zero-shot. The variation implies a non-uniform transfer of detection and generation skills. Feature analysis revealed systematic differences, particularly attribution markers ("said", 0.0537 importance) and structural elements ("featured", 0.0260), reflecting GPT-4o's training on journalistic text rather than authentic misinformation patterns. Such stylistic differences concur with Chang et al. (2024), who discovered similar lexical distinctions between AI-generated and human-authored misinformation.

There are some experimental flaws that should be talked about: using a single-source dataset (ErfanMoosaviMonazzah) may have had biases that were specific to that domain; splitting samples evenly between categories (3,000 samples per category) may not have shown how misinformation ratios work in the real world, where information about politics is much more common than information about other topics; and hardware limitations that made transformer training apply to two epochs may have made the model look worse than it really was. The study reveals intriguing prospects for the advancement of fake news detection systems, particularly the potential enhancement of solutions through the integration of ALBERT's parameter efficacy with probabilistic methodologies.

7 Conclusion and Future Work

This research investigated the central question: **How can the integration of GPT-4o's generative capabilities with ALBERT's classification framework improve fake news**

detection performance across diverse content domains while maintaining adaptability to evolving misinformation tactics?

Even though the blend didn't make detection more accurate as we had hoped, the paper gave us useful information that changed the way we think about synthetic data in adversarial situations. The paper did show that LLMs can make a lot of convincing false information, but it also showed that there are big differences between false information made by AI and false information made by people that can't be directly transferred.

7.1 Key Findings

The research accomplished three goals, each with different results. It first successfully created and used advanced prompting methods to make 15,000 high-quality fake news samples with a 95.2% success rate. Next, instead of showing how to improve cross-domain performance, the experiments showed a 27.6% mean decline, which was a "productive failure" that showed important differences between fake and real-world misinformation. Thirdly, the combined system only got 68.9% right (short of the 85% goal), but it did find ALBERT to be an extremely strong architecture that kept 99.5% accuracy through distribution shifts.

The most significant finding from that research is the asymmetric generation-detection relationship: although GPT-4o effectively produced evasive misinformation, detectors trained on synthetic data exhibited poor performance in real-world detection (53.3% accuracy). The asymmetry does not indicate failure; rather, it illuminates the nature of misinformation and the constraints of contemporary AI.

7.2 Contributions and Impact

This research has some important findings, even though it didn't meet traditional standards of success. Cross-evaluation framework offers a structured approach to measure the quality of synthetic data in challenging environments, which is essential given the rising utilisation of synthetic data. A scalable generative pipeline (312 samples/hour) and strong parsing methods give researchers tools they can use in future studies. Most importantly, studies provide experimental evidence that robustness in detection is not achieved through an increase in training data volume, but rather through a comprehension of the intrinsic differences between human and AI-generated content, rather than merely augmenting training data volume.

The discovery that ALBERT demonstrates unexpectedly high robustness (0.690 score) indicates that parameter-efficient models provide unanticipated resilience in adversarial contexts and open new avenues for model development research. The unveiled explicit feature analysis, which reveals discriminative markers, provides pragmatic recommendations for the enhancement of generation and detection methodologies.

7.3 Future Directions

This work lays the groundwork for a number of potentially interesting future projects, such as adversarial co-training, in which the generator and detector models learn from each other; combined systems that combine ALBERT's simplicity with probabilistic methods; multi-modal detection using metadata and propagation dynamics; and human-in-the-loop systems that use the best of both AI and human intuition. Fine-tuning methods, within the limits of what is seen as morally acceptable, may produce synthetic samples that look real and can be used to train strong detectors.

7.3.1 Adversarial Co-training

The asymmetrical synthetic datasets found show a 27.6% evasion success rate but only a 53.3% training effectiveness rate. This means that the current generation learns how to evade detection on the surface rather than how to deceive on a deeper level. Adversarial co-training and reinforcement learning should close the gap by building dynamic systems that find the right balance between detection evasion and training usefulness.
Adversarial co-training would create a continuous feedback loop between the generator and the detector, which is different from the fixed generation strategy used here. The generator would be rewarded not only for avoiding detection but also for producing samples that improve detection accuracy when used as training data for real fake news. This two-task goal would move the system away from the journalistic tendencies of the GPT-4o and towards real misinformation traits. ALBERT is a great discriminator because it is stable (99.5% accuracy preserved under distribution shifts) and gives stable gradient signals that guide generation towards real deception patterns instead of just copying styles.
This would be further supported by conceptualising deception trait acquisition as a sequential decision-making challenge with a multi-faceted reward system. The reward function would be designed to target the researched vulnerabilities: penalising excessive reliance on attribution markers (with "said" rated as 0.0537 important) and incentivising unattributed statements and emotional manipulation tactics commonly associated with viral misinformation. A curriculum learning approach would gradually add complexity, starting with simple factual manipulations and moving up to more complex psychological manipulations and real deception skills without mode collapse.

References

Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “ALBERT: A lite BERT for self-supervised learning of language representations,” arXiv (Cornell University), Jan. 2019, doi: 10.48550/arxiv.1909.11942. Available: https://arxiv.org/abs/1909.11942

A. Singh and S. Patidar, “A Survey on Fake News Detection Using Machine Learning,” IEEE, pp. 327–331, Dec. 2022, doi: 10.1109/icac3n56670.2022.10074450. Available: https://doi.org/10.1109/icac3n56670.2022.10074450

C. Agrawal, A. Pandey, and S. Goyal, “A survey on Role of machine learning and NLP in fake news detection on social media,” 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), vol. 1319, pp. 1–7, Sep. 2021, doi: 10.1109/gucon50781.2021.9573875. Available: https://doi.org/10.1109/gucon50781.2021.9573875

A. Narkhede, D. Patharkar, N. Chavan, R. Agrawal, and C. Dhule, “Fake News Detection Using Machine Learning Algorithm,” IEEE, pp. 166–170, Nov. 2023, doi: 10.1109/iccsai59793.2023.10421153. Available: https://doi.org/10.1109/iccsai59793.2023.10421153

M. Iftikhar and A. Ali, “Fake News Detection using Machine Learning,” IEEE, pp. 103–108, Feb. 2023, doi: 10.1109/icai58407.2023.10136676. Available: https://doi.org/10.1109/icai58407.2023.10136676

J. Jouhar, A. Pratap, N. Tijo, and M. Mony, “Fake News Detection using Python and Machine Learning,” Procedia Computer Science, vol. 233, pp. 763–771, Jan. 2024, doi: 10.1016/j.procs.2024.03.265. Available: https://doi.org/10.1016/j.procs.2024.03.265

A. Pal, N. Pranav, and M. Pradhan, “Survey of fake news detection using machine intelligence approach,” Data & Knowledge Engineering, vol. 144, p. 102118, Nov. 2022, doi: 10.1016/j.datak.2022.102118. Available: https://doi.org/10.1016/j.datak.2022.102118

B. Palani and S. Elango, “BBC-FND: An ensemble of deep learning framework for textual fake news detection,” Computers & Electrical Engineering, vol. 110, p. 108866, Jul. 2023, doi: 10.1016/j.compeleceng.2023.108866. Available: https://doi.org/10.1016/j.compeleceng.2023.108866

M. Q. Alnabhan and P. Branco, “Fake News Detection Using Deep Learning: A Systematic Literature Review,” IEEE Access, vol. 12, pp. 114435–114459, Jan. 2024, doi: 10.1109/access.2024.3435497. Available: https://doi.org/10.1109/access.2024.3435497

M. F. Mridha, A. J. Keya, Md. A. Hamid, M. M. Monowar, and Md. S. Rahman, “A comprehensive review on fake news detection with deep learning,” IEEE Access, vol. 9, pp. 156151–156170, Jan. 2021, doi: 10.1109/access.2021.3129329. Available: https://doi.org/10.1109/access.2021.3129329

Q. Chang, X. Li, and Z. Duan, “Graph global attention network with memory: A deep learning approach for fake news detection,” Neural Networks, vol. 172, p. 106115, Jan. 2024, doi: 10.1016/j.neunet.2024.106115. Available: https://doi.org/10.1016/j.neunet.2024.106115

E. Papageorgiou, C. Chronis, I. Varlamis, and Y. Himeur, “A survey on the use of Large Language Models (LLMs) in fake news,” Future Internet, vol. 16, no. 8, p. 298, Aug. 2024, doi: 10.3390/fi16080298. Available: https://doi.org/10.3390/fi16080298

S. Kuntur, A. Wróblewska, M. Paprzycki, and M. Ganzha, “Under the Influence: A survey of large language models in fake news Detection,” IEEE Transactions on Artificial Intelligence, pp. 1–21, Jan. 2024, doi: 10.1109/tai.2024.3471735. Available: https://doi.org/10.1109/tai.2024.3471735

OpenAI, “GPT-4O System Card,” Aug. 2024. Available: https://cdn.openai.com/gpt-4o-system-card.pdf

K. I. Roumeliotis, N. D. Tselikas, and D. K. Nasiopoulos, “Fake News Detection and Classification: A comparative study of convolutional neural networks, large language models, and natural language processing models,” Future Internet, vol. 17, no. 1, p. 28, Jan. 2025, doi: 10.3390/fi17010028. Available: https://doi.org/10.3390/fi17010028