National College of Ireland

Detection of AI-Generated Fake Reviews: A Comparative Analysis of Transformer and Recurrent Architectures Across Language Model Generations

MSc Research Project

MSC Data Analytics

SAI KALYAN BADE

Student ID: X23388111

School of Computing

National College of Ireland

Supervisor: Arghir Nicolae Moldovan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: SAI KALYAN BADE

Student ID: X23388111

Programme: MSc Data Analytics **Year:** 2026

Module: MSc Research Practicum

Supervisor: Arghir Nicolae Moldovan

Submission Due Date: 28/01/2026

Project Title: Detection of AI-Generated Fake Reviews: A Comparative Analysis of Transformer and Recurrent Architectures Across Language Model Generations

Word Count: 9576 **Page Count:** 24

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: SAI KALYAN BADE

Date: 28/01/2026

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
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.	□

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):	

Detection of AI-Generated Fake Reviews: A Comparative Analysis of Transformer and Recurrent Architectures Across Language Model Generations

Sai Kalyan Bade

X23388111

Abstract

The increased utilization of advanced language models has enabled the creation of various fake product reviews, which could adversely affect consumer trust and the credibility of e-commerce platforms. The proposed work investigates the detection of AI-written fake reviews by comprehensively evaluating four detection models on various generations of language models and product types. The study evaluated the Attention-GRU, RoBERTa, DeBERTa, and XLNet models on two datasets: the reviews written by the GPT-2 across ten Amazon products, including FraudSquad reviews written by the three modern language models Qwen-72B, LLaMA-3-8B, and DeepSeek-R1-Qwen across seven categories. The experimental result revealed that the current fine-tuned language models produce more detectable content than their predecessors, GPT-2, which negates the presumptive hypotheses stating the greater difficulty level of detection represented by newer LLMs. Transformer models have always given more accurate results than the recurrent models. For instance, DeBERTa and RoBERTa achieved 99-100% accuracy against modern LLM content, while Attention-GRU achieved only 88-96%. These models of detection suggested good generalization when there was a significant difference in the categories with the total error of only 2 to 5 points between the two kinds of products. The paper is the first to systematically compare the ability of different generations of language models to detect fake reviews gained through AI and corroborates the capacity of current transformer-based detection models to detect high-quality fake reviews with near-perfect precision, thus offering valuable lessons to approaches with e-commerce platform protection.

1 Introduction

1.1 Background

Online product information regarding the reviews has become primary in consumer choice in e-commerce. Studies show that more than 90 percent of buyers consider reviews prior to making a purchase, positive reviews have the potential to boost the conversion rate by 270 % (Mohawesh et al., 2024). The worldwide e-commerce sector amounted to more than 5.8 trillion USD in 2023, with authenticity of reviews being the key to preventing a gap in the market integrity (Statista, 2024). This reliance on user-made reviews exposes it to manipulation using false reviews.

Large language models have altered the environment of fake reviews. The initial models such as GPT-2 (Radford, 2019) had the ability to generate automated review, where human judges did not tell the difference between the text generated by the machine and the real content (Salminen et al., 2022). The modern models such as Qwen-72B, LLaMa-3, and DeepSeek are major improvements in the quality of generated text, they generate more and more human-like content and develop complex reasoning patterns. These new fine-tuned models are able to produce reviews that are contextually relevant on a large scale being unprecedented challenges on detection systems (Wang et al., 2024). The spread of this kind of AI generated fake content poses a threat to consumer confidence, good market competition and integrity of the e-commerce platforms.

1.2 Novelty and Contributions

1.2.1 Project Novelty

This study is a first systematic comparative study of detection potential of language models of various generations with reference to e-commerce review fraud. As opposed to the past research, which aims at a single model type or older generation AI models, this paper discusses both legacy (GPT-2) and newer fine-tuned models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen) to learn about changing detection issues. This work presents a new evaluation framework, which detects product types in seventeen categories, the first work to offer the cross-domain generalization analysis at the largest scale.

Key aspects of this study's novelty include:

- Systematic comparison across GPT-2 (old) and modern fine-tuned LLMs (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen)
- Comprehensive architectural diversity: Recurrent (Attention-GRU) and Transformer-based (RoBERTa, DeBERTa, XLNet) models
- Large-scale cross-domain evaluation: 17 product categories with consistent evaluation framework
- First work to analyze detection generalization capabilities across different product types and LLM generations

1.2.2 Research Contributions

The main contributions to the research are:

- **Comprehensive multi-generation analysis:** The original comparison of detection performance between GPT-2 and three modern fine-tuned language models that first compares the use of the latter and the former in a systematic way, indicating that the modern models generate more detectable features than is initially expected.
- **Architectural performance benchmarking:** Comparative analysis of four distinct detection architectures (Attention-GRU, RoBERTa, DeBERTa, XLNet) on the same data sets, that makes obvious vigorous performance arrangements and trade-offs between them.
- **Cross-category generalization assessment:** Cross category checking between seventeen different product categories with strong generalization with differences of no more than 2-5 percentage points, which proves the implementation of unified detection systems.
- **Near-perfect detection achievement:** It has been demonstrated to achieve 99-100% accuracy rates on transformer models on modern LLM content, which were the current limits of highest detection performance.

- **Computational efficiency analysis:** Characterised what accuracy-efficiency trade-off exists between the recurrent applications (88-96% accuracy) and transformer application (97-100% accuracy) conveying the deployment choice.
- **Anomaly discovery:** the architecture-related vulnerabilities, especially the 7-percentage points of performance-degradation of XLNet on DeepSeek-R1-Qwen content, which point to the possible security issues.
- **Practical implementation framework:** Design of e-commerce platform deployment strategies that trades off between detection quality and computing power.

1.3 Motivation

The current studies of fake review detection have concentrated majorly on human-written deceptive reviews or data of older language models. There is a significant understanding gap to the effectiveness of detection systems that is faced when reviews are presented by different generations of language models. The new large language models have varied linguistic properties or features than the old models, and may necessitate distinct detection strategies. Wang et al. (2024) obtained a detection accuracy of 99.72 to determine that AI-generated text could be identified with the help of BERT though the evaluation employed a single set of data that was not compared across models.

Also, there has been no comprehensive systematized comparison of detection architectures in the efficiencies and deployment arena. Transformer-based models RoBERTa, DeBERTa, and XLNet are also shown to be useful in the natural language understanding field, with DeBERTa achieving 95.2% accuracy on fake review prediction (Hou et al., 2025). The approach of using recurrent architectures with capabilities of attention has potential benefits to efficiency particularly in the case of application to resource-constrained deployment. The e-commerce platforms Enforce sometimes need to be implemented with regard to the selection of methods of detection to practice striking a balance between efficiency and feasibility of operation. Extrapolation of the models of review detections to new categories has not been given the proper consideration. Moreover, the generalization capabilities of the detection models in terms of different product categories have not been well-researched although the contents of the product review are different.

1.4 Research Questions

The study examines AI-generated fake review detection, utilizing three major questions of research:

RQ1: How does detection accuracy vary across different language model generations? This query looks into whether the reviews produced by current fine-tuned language models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen) are more challenging to detect than the ones produced by similar previous models, like GPT-2.

RQ2: Which detection architecture provides optimal effectiveness for fake review identification? This query undertakes comparisons between less difficult recurrent solutions incorporating attention-enhanced GRU and transformer-based solutions, such as RoBERTa, DeBERTa, and XLNet.

RQ3: Do detection models generalize consistently across different e-commerce product categories? This question examines the hypothesis of whether product type has a significant variation in performance or not, irrespective of review domain.

1.5 Objectives

The study has four distinct objectives that should be accomplished:
• **Objective 1:** Establish baseline detection performance on GPT-2 generated reviews across ten product categories using four distinct detection architectures
• **Objective 2:** Evaluate the same detection models on three modern language models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen) across seven product categories to assess how detection performance evolves with newer LLM generations
• **Objective 3**: Analyse cross-category generalization capabilities by examining whether detection model performance remains consistent across different e-commerce product types
• **Objective 4**: Achieve robust detection performance (>90% accuracy) across all datasets and product categories while providing architectural recommendations for practical deployment in e-commerce systems

1.6 Methods Overview

The research is an experimental comparative study based on two major datasets. The initial one is the dataset of Salminen et al. (2022) that consists of the generated review texts by the GPT-2 algorithm and was limited to ten categories on the Amazon product review platform. The second one is the FraudSquad review data set produced by three state of the art language models and comprising seven classes. Four detection architectures are examined, including a bidirectional GRU architecture that uses the additive attention, three transformer-based models (RoBERTa, DeBERTa, and XLNet models), which are fine-tuned on the binary classification setup. There is accuracy, precision, recall and F1-score performance measures that are evaluated comprehensively in terms of categories.

1.7 Report Structure

The remainder of the report has a logical structure. The theoretical background described in the literature review provides a basis and outlines the gaps that are addressed in the current study. The chapter on methodology presents the datasets, designs of the models, and evaluation procedures. The design specification describes the system architecture and connection among the components. The implementation is based on the technical implementation of the new approach. Assessment presents experimental results on different datasets and settings and conclusions are made according to the situation of the research question. Of importance, in the conclusion part, there is a discussion of significant findings and work forward.

2 Related Work

2.1 Overview

As e-commerce websites are more prone to manipulation, the research about the ability to detect fake reviews and text produced by AI has gained a huge significance. The literature review in this study is relevant and falls into the following categories whereby detection approaches are based on transformer architecture models, recurrent neural networks, and graph-based models. The significance and drawbacks of the available literature have been identified in this discussion as the reason behind the present study. The table 1 below summarizes the applicable experimental research in this field.

2.2 Transformer-Based Detection Approaches

Transformer architectures (Vaswani *et al.*, 2017) have become the primary method for text classification tasks, such as identifying fake reviews. Mohawesh et al. (2024) suggested a mixed method for finding fake reviews that used both RoBERTa and LSTM layers. The system got 96.03% right on the OpSpam dataset and 93.15% right on the Deception dataset. The hybrid architecture does a good job of capturing complex linguistic patterns in fake reviews by combining transformer contextual representations with sequential modeling. But the method needs a lot of computer power, which makes it hard to use in real-time situations. The study confirms the efficacy of transformers; however, it does not specifically tackle the detection of AI-generated content.

Wang et al. (2024) used BERT-based classification to identify AI-generated text, achieving a high accuracy rate of 99.72% with training loss close to 0.021. This is evidence enough to prove the efficiency and ability of transformer models to differentiate between AI-generated texts and writings by human beings. The weakness of the study is based on the consideration restricted to a singular dataset and the inability to contrast various generations of language models. The study proves the transformers' ability to detect AI-generated texts but fails to provide systematic comparisons to support implementation.

Hou et al. (2025) conducted a study on context-aware models for identifying fraudulent reviews and analyzed DeBERTa, ELECTRA, and RoBERTa on a joint Amazon and Yelp corpus of 100,000 reviews. DeBERTa gave the best result among the rest, with accuracy of 95.2%. This is due to the fact that DeBERTa has an attention mechanism which decouples content and position encoding. The context-aware model is very effective when it comes to identifying deception patterns at a very nuanced level. Nevertheless, the assessment is conducted on a long-existing fake review and not on the generated content produced by modern language models. This research proves DeBERTa is superior to other versions of transformers but doesn't tackle the issue of identifying AI-generated contents.

Clark et al. (2020) have developed ELECTRA, which has pretraining through discriminators rather than masked language modeling. The model achieved the highest score in SQuAD 2.0 and 91.1% in GLUE benchmark and involved less samples compared to other approaches. The discriminator training paradigm can be applied in the detection of real and fake text. The publication creates a theoretical framework of detection methodologies but does not address, in particular, the issue of the classification of fraudulent reviews.

2.3 Recurrent Neural Network Approaches

Another option that can be used to substitute transformers is recreational architectures, and this may lead to computational advantages. The solution provided by Padalko et al. (2024) was created to detect fake news based on different combinations of bidirectional LSTM and GRU with attention mechanisms. The solution had a correct answer to the majority of questions (90) of the ISOT and FA-KES dataset. The mechanism of attention brings about maximum efficiency to the solution as it enables the solution to pay attention to the areas of the text most relevant to it. To the extent that it is not as accurate as the solution of transformers, it is as a compromise between speed and accuracy. This paper demonstrates that attention-based recurrent architecture receives high efficiency in solving problems with a limited number of resources.

Zhang et al. (2022) proposed Bi-GRU using keywords attention (REKA) in relation extraction and achieved a wide range of F1-score (84.8). The keywords attention mechanism is performance enhancing and requires no external tools in natural language processing. Focused attention technique is successful in identifying some text patterns that have significance on classification. Its limitation lies in the fact that it is specifically applied to relation extraction,

rather than counterfeit review detection. The study affirms the usefulness of GRU-attention structures in the task of classifying text.

Liu et al. (2022) created a hierarchical architecture, in which the BiGRU, attention, and convulations layers are added to detect spam data in e-commerce data sets like AGNews, IMDB, and THUCNews. This finding was better when compared to other baselines since the outcome of the multi-scale attention mechanism of N-gram convolutions and hierarchical features was better. Such architecture is a variant of the integration of the merits of recurrent and convolutional parts at the same time to enhance spam detection. The fact that the architecture is so complicated may serve as an impediment to flexibility. This paper revisits the effectiveness of the hybrid RNN-CNN architecture with regard to review classification.

Arunkrishna and Senthilkumaran (2025) proposed BiLSTM-GRU attention hybrid model with which they could detect fake news in Tamil and English with accuracy of 90.6 percent in emotion classification. The hybrid architecture which integrates both LSTM and GRU bidirectionally and uses attention works best when there are many languages involved. The investigation depicts the generalization between languages as the focus is put on news as opposed to product reviews. The study supports the effectiveness of attention-enhanced recurrent architectures on deception detection activities.

2.4 Graph Neural Network Approaches

Graph methods further provide additional data as they are used to model the relationships between reviews, consumers and products. Li et al. (2019) developed an anti-spam (GAS) system based on the idea of graph convolutional networks (GAS) that used GraphSAGE and GCN on the data of Xianyu million-scale platform. Heterogeneous and homogeneous graphs are effective to reflect global and local semantics of review data. The methodology entails the users to describe a powerful graph framework of the interconnections of the users, products, and reviews. The literature demonstrates the possibility of learning graphs, however, it will only work provided that the data is represented dissimilarly not only with texts.

Duma et al. (2024) conducted a systemic literature review on the use of graph neural networks to identify fake review and investigated approaches that combine various GNNs with BERT, resulting into an accuracy of 91.2. Graph learning effectively represents more complex associations between reviews, users and products that text-only techniques cannot describe. The relationship data are not available hence making it difficult to use when the relationship data and their requirements are complicated. The review highlights complementary avenues and touchpoints but fails to mention AI -generated content specifically.

2.5Synthesis and Research Gap

The literature shows that there are a number of consistent patterns across different detection methods. Transformer-based models consistently achieve accuracy exceeding 95% on fake review detection tasks, with DeBERTa and RoBERTa demonstrating particularly robust performance. Recurrent architectures with attention mechanisms are 85–90% accurate and are also more efficient in terms of computing power. Graph-based methods give extra signals from relational patterns, but they need more data than just the review text.

Although much progress has been made, it is apparent that three large gaps are currently not filled. First, there is a gap regarding the comparison between the detection capabilities of various generations of language models. Research evaluates either the output of earlier versions of GPT-2 models or assumes identical dynamic responses of any form of AI-generated texts. The shift from GPT-2 towards more modern versions such as Qwen, LLaMA, and DeepSeek could raise unique detection difficulties that requires more analysis. Second, it is also observed that systematic comparisons involving various detection architectures on common datasets are not conducted. In most studies, either one or group of them are evaluated which are insufficient

to understand the comparative capabilities of the detectors. Third, the aspect of category-wise generalization is not yet explored. E-commerce platforms contain reviews of products belonging to various categories. Not much work has been done on analysing the capabilities of detection models on various categories and their possible domain gaps.

This study addresses all three gaps by systematically comparing four detector models (Attention-GRU, RoBERTa, DeBERTa, XLNet) in the two datasets that consider multiple model generation language models, as well as seventeen product categories. The analysis provides practical guidance on e-commerce platforms that are interested in putting in place effective detection mechanisms.

Table 1: Literature Summary - Artificial Intelligence Generated Text and Fake Review Detection Experimental Studies.

Study	Dataset	Models Tested	Best Performance	Key Finding
Mohawesh et al. (2024)	OpSpam, Deception	**RoBERTa-LSTM,** BERT, CNN	96.03% accuracy	Hybrid transformer-LSTM captures intricate patterns
Wang et al. (2024)	**AI-generated text**	**BERT**	**99.72% accuracy**	**BERT effective for AI text detection**
Hou et al. (2025)	Amazon, Yelp (100K)	**DeBERTa**, ELECTRA, RoBERTa	95.2% accuracy	DeBERTa outperforms other transformers
Padalko et al. (2024)	ISOT, FA-KES	**BiLSTM-BiGRU** with attention	90% accuracy	Attention enhances RNN performance
Li et al. (2019)	Xianyu (million-scale)	**GCN, GraphSAGE**	Superior to baselines	Graph captures review-user-product context
Zhang et al. (2022)	E-commerce reviews	**Bi-GRU with attention**	84.8% F1-score	Keywords attention improves extraction
Liu et al. (2022)	AGNews, IMDB	**BiGRU-Attention-C**NN	Superior to baselines	Multi-scale attention improves spam detection
Duma et al. (2024)	Amazon, Yelp, TripAdvisor	**GNN with BERT**	91.2% accuracy	Graph learning captures complex relationships
Clark et al. (2020)	SQuAD 2.0, GLUE	**ELECTRA**	91.1% GLUE	Discriminator pretraining more efficient

Note: The performance (99.72% accuracy) of Wang et al. (2024) in the table was the highest, and that is why this row is bolded.

3 Research Methodology

3.3 Research Approach

In this work, the experimental comparative work is used and the results of fake review detection are evaluated according to different language model generations and different detection architecture. A team of four detection models is going to be systematically evaluated on two main datasets with the help of the study design, which will allow direct performance traits comparison. The procedure is based on a quantitative framework to examine the accuracy

of classification and other contingent measures in the various types of products. The experimental design controls confounding variables by controlling all preprocessors, training protocols of all the types of architectures and evaluation protocols. Known random seeds and recorded settings ensure that it is possible to repeat the results. The general workflow of the methodology is presented as Figure 1.

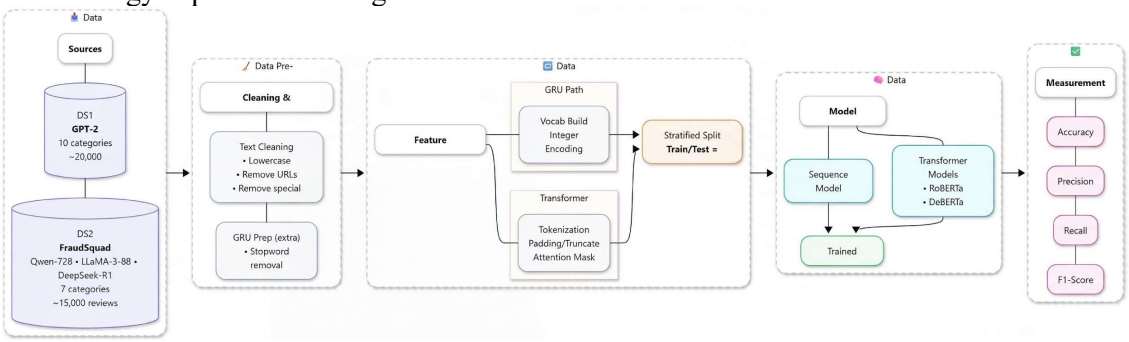

Figure 1 overall methodology workflow

3.4 Datasets

3.4.1 Dataset 1: GPT-2 Generated Reviews

The first dataset is from Salminen et al. (2021), which has real reviews by people and reviews generated by the GPT-2 language model. This dataset, publicly available at https://github.com/joolsa/FakeReviews, has information regarding ten various kinds of products sold on the Amazon website, such as books, clothes, shoes, and jewelry; electronics; home and kitchen; the Kindle Store; movies and TV; pet products; sports and outdoors; tools and home improvement; and lastly, toys and games. There are 20,000 reviews, including a combination of computer-generated (CG) and original real (OR) reviews in each group. This dataset is used to identify the efficiency of older language model-generated contents.

3.4.2 Dataset 2: FraudSquad - Modern Language Models

The second is the FraudSquad Dataset (Liu et al., 2025), which is comprised of Amazon 2022 5-core reviews and content generated by three modern fine-tuned language models. This dataset, publicly available at https://anonymous.4open.science/r/FraudSquad-5389, includes seven categories of products. Everything ranging from beauty products to appliances, baby products, health and personal care, musical instruments, software, and finally, video games. Four various versions of this dataset were tested:
There are 4,982 samples in the Qwen-72B subset, and they all contain reviews written by the Qwen-72B language model and also contain real reviews. There are 4,990 samples in the LLaMA-3-8B subset, and they were generated by the LLaMA-3-8B model. There are 4,990 samples in the DeepSeek-R1-Qwen subset, and these samples were generated by the DeepSeek-R1-Qwen model. The total number of samples to test cross-model generalization is approximately 14,970 samples, which is the combination of the three language model subsets.

3.4.3 Data Partitioning

All the datasets were exposed to identical partition procedures. The distribution between the training and test datasets is 80:20. Stratified sampling ensured equal representation of computer-generated and original reviews in both groups. The fixed random seed value, 2021,

ensured the ability to repeat all experiments. Data augmentation wasn't used to conform to standard methods of evaluation.

Table 2: Dataset Summary

Dataset	LLM Source	Categories	Total Samples	CG Samples	OR Samples
DS1	GPT-2	10	40,433	20,238	20,216
DS2.1 Qwen	Qwen-72B	7	4,982	2,495	2,487
DS2.2 LLaMA	LLaMA-3-8B	7	4,990	2,495	2,495
DS2.3 DeepSeek	DeepSeek-R1-Qwen	7	4,990	2,495	2,495
DS2.4 Combined	All 3 LLMs	7	14,970	7,485	7,485

3.5Model Architectures

Four detection architectures were evaluated, with configurations summarized in Table 3. The **Attention-GRU** model employs bidirectional recurrent layers with an additive attention mechanism (Bahdanau, Cho and Bengio, 2014) that learns alignment scores between hidden states and sequence positions, enabling focused context aggregation for classification. The vocabulary is built from training data, limited to the most frequent terms.

The **RoBERTa(Liu *et al.*, 2019)** architecture applies robustly optimized BERT pretraining with byte-pair encoding tokenization. A classification head converts pooled representations into binary predictions. Fine-tuning adapts pretrained weights to the review classification task.

The **DeBERTa** (He *et al.*, 2020) version 3 model introduces disentangled attention that separately encodes content and position information, enabling more nuanced attention computations. Gradient-disentangled embedding sharing improves training stability compared to standard approaches.

The **XLNet** (Yang *et al.*, 2019) architecture uses permutation language modeling to capture bidirectional context through autoregressive factorization without masking corruption. Left-side padding aligns with the autoregressive generation direction, distinguishing it from other transformer variants.

Model choice: These four architectures were selected to cover both lightweight recurrent models and large transformer-based detectors: Attention-GRU represents an efficient baseline suitable for constrained environments, while RoBERTa, DeBERTa and XLNet are state-of-the-art transformer models commonly used in text classification, allowing a fair comparison between efficiency-oriented and accuracy-oriented designs.

Table 3: Model Configuration Comparison

Model	Architecture	Layers	Hidden Size	Sequence Length	Parameters
Attention-GRU	BiGRU + Attention	2	256	1,000	2M
RoBERTa	Transformer	12	768	256	125M
DeBERTa	Disentangled Transformer	12	768	256	184M
XLNet	Permutation Transformer	12	768	256	117M

3.6Training Configuration

All of the experiments were done in environments that used GPUs to speed up computing. The Adam optimizer took care of updating the parameters for all of the models. The binary cross-entropy loss showed how wrong the classification was.

Transformer models had the same hyperparameters: training lasted for one epoch, with a batch size of 8 and a learning rate of 1e-5. The single epoch was enough because the initialization

was done ahead of time and the datasets were not very big. Standard backpropagation is used without accumulation to find the gradients.
GRU model had to be configured differently as it was trained using random initial values: 10 epochs and a batch size of 128 with a learning rate of $3e^{-4.}$ The gradient clipping helped in preventing blow up of gradients through its maximum value of 5. The checkpoint that worked best according to its validation set performance was retained to the last evaluation.
To achieve reproducibility all the tests used fixed random seeds (value: 2021, selected arbitrarily) of the random number generators in Python, NumPy and the PyTorch. CUDA deterministic operations ensured that repeatability of operation was possible.

Accuracy was used as the primary metric for comparability with prior work, while precision, recall and F1-score were included because they capture the trade-off between false alarms (flagging real reviews as fake) and missed detections (failing to catch AI-generated reviews), which is critical for e-commerce platforms.Confusion matrices and per-category statistics were further analyzed to understand error patterns across product types and language model generations, aligning directly with the research questions on cross-category and cross-generation generalization.

3.7 Evaluation Metrics

There are several different metrics that were used during performance assessment upon which they worked very well. The primary metric of comparison was accuracy which checked the percentage of correct predictions on all the samples. Precision was determined by the number of the predicted positives that were true positives, and this indicated the number of false positives. Recall was used to determine the number of the real positives selected right, indicating the false negative levels. The harmonic mean of the precision and recall identified the F1-score because it was fair in terms of the two kinds of errors (Sokolova and Lapalme, 2009).
Confusion matrices clearly gave details on the breakdown of errors per model-dataset combination. The analysis was done in categories in order to determine the extent to which it was generalized. Mean accuracy and standard deviation by category were used to describe the overall behaviour of the model. Cross-dataset studied the differences in the pattern of performance between the GPT-2 and modern language models text.

4 Design Specification

4.3 Overall System Architecture

The detections system is designed with a modular pipeline design comprising of four key phases namely, data ingestion, preprocessing, model training, and evaluation. In this design, components of the system can be swiped without altering interfaces of the stages.
The stage of data ingestion is the one that is responsible to load review datasets based on structured files. Raw data is inputted to the processing stage of preprocessing that includes text and label data. The preprocessing phase transforms raw text to model ready representations. Transformer architecture and recurrent architecture have two directions. At step of training, model parameters are optimized with the help of pre-processed data under the supervisory learning. The evaluation phase is carried by means of trained models on previously unutilized test data and is used to compute performance measures and produce analysis output.

With training, the only direction the data moves is up the pipeline. The evaluation phase during the inference process does not involve training. This separation allows one to perform batch evaluations with rapidity as they do not require retraining.

4.4 Data Processing Pipeline

4.4.1 Transformer Models Pipeline

Transformer preprocessing pathway involves the use of pretrained model specific tokenizers to shred the text into tokens. The text input is divided into subwords creating token sequences. In the event that the sequence is longer than the maximum length (256 tokens) it is trimmed at the end. In order to ensure that the sizes of all the batches are similar, the shorter sequences receive padding tokens.

Attention masks are used together with token sequences and indicate which tokens are legal and which are padding positions. The masks allow the models to disregard padding whenever they are calculating attention. To calculate the loss, source labels are transformed into binary integers (CG/OG) to perform loss calculation.

The tokenization process preserves the original meaning of a text but modifies it into numbers. In the start of a sequence, it has classification token and the separator token is found at the end of the sequence. These tensors have token identifiers, attention masks and target labels that can be used by the model.

4.4.2 GRU Model Pipeline

The preprocessing of the recurrent model also takes a different direction that is concerned with assembling new vocabulary. Text cleaning also converts all the letters to lowercase, resizes the abbreviations and removes URLs, usernames and special characters. Stopword filtering removes words that are common parents that do not bring much sense to the table. Lemmatization splits words into basic ones to assist in vocabulary construction.

The vocabulary is constructed solely on the basis of the training set and prevents information to leak over to the test set. This vocabulary is composed of the 1,000 most frequent tokens, and the rest of the words not represented in the vocabulary are mapped to a special unknown token. The tokens are encoded into vocabulary indexes with the use of integer encoding.

In sequence padding, the information is pushed to the right side of a sequence at the end. This conformation coincides with the GRU processing data, in which enduring final hidden states generalize the sequence text. The specific length of a sequence (1, 000 tokens fixed) permits varying length of review.

4.5 Model Training Framework

The training framework uses supervised learning by optimizing it over and over again. Every training epoch works on the whole training set in groups. Batch processing makes it possible to use the GPU more efficiently by allowing multiple computations to happen at once.

The forward pass uses input representations to make predictions about the model. The pretrained encoder makes contextual embeddings for transformer models, and the classification head makes logits. In the GRU model, hidden states are made by processing data in order through recurrent layers, context vectors are made by attention weighting, and predictions are made by the output layer.

Loss computation uses binary cross-entropy to figure out how wrong a prediction was. The loss measures how far off the predicted probabilities are from the true labels. Backpropagation is used to calculate gradients, which then tells the model which parameters to change. The optimizer uses gradients to change the weights of the model, and the learning rate controls how much the weights change.

The progress monitoring in training maintains a check over the loss and accuracy periodically. In the case of transformer models, it is common to only train the model once since it is already trained. To prevent overfitting in GRU model, use several epochs and select the best checkpoint in terms of its performance on the validation.
In model persistence, the trained parameters are saved in such a way that they can be accessed in the future. Transformer models store their settings/weights in a standard manner. GRU model maintains state dictionaries, as well as vocabulary mappings.

5 Implementation

5.3 Development Environment

The tool used Python as the programming language as it has numerous machine learning and natural language processing tools. The deep learning framework called PyTorch enabled carrying out the computations and automatic differentiation of tensors to ensure gradient-based optimization. The ability of CUDA to execute the training of big neural network models on GPUs enabled them to train them in a short time.

The Hugging Face Transformer library provided the pretrained transformers models, along with the corresponding tokenizers. The Hugging Face Transformer library standardizes model loading, fine-tuning, and works the same way across various models. The Pandas library handled the tabular data operations necessary for data loading and result integration. NumPy provided the capabilities to perform array operations, which were necessary for data transformation. The library, Scikit-learn, provided capabilities to perform splitting operations on data and evaluation metric definitions. The Natural Language Toolkit library (NLTK) helped the recurrent model handle the text preprocessing operations such as the formation of sets of stopwords and lemmatization.

Python, PyTorch and Hugging Face Transformers were chosen as the core technology stack because they provide mature, well-supported libraries for deep learning and NLP, extensive community-validated model implementations, and straightforward reproducibility through shared code and pretrained checkpoints.Supporting libraries such as Pandas, NumPy, Scikit-learn and NLTK were used because they are de-facto standards for data manipulation, evaluation and text preprocessing in machine learning workflows, simplifying the replication of this study by other researchers and practitioners.

Table 4: Implementation Framework Summary

Component	Technology	Purpose
Language	Python	Core development
Deep Learning	PyTorch	Neural network training
Transformers	Hugging Face	Pretrained models
Data Processing	Pandas, NumPy	Data manipulation
Evaluation	Scikit-learn	Metrics computation
Text Processing	NLTK	GRU preprocessing
Acceleration	CUDA	GPU computation

5.4Model Implementations

5.4.1 Transformer Model Implementation

The transformer models used weights that had already been trained from the Hugging Face model repository. When the model was started, it loaded base architectures (RoBERTa, DeBERTa, XLNet) with sequence classification heads set up to give binary output. The

classification head turns pooled representations into two-dimensional logits that show the chances of each class.
Tokenizers matched each model architecture to make sure that text encoding and model expectations were the same. Tokenization created sequences of a fixed length by cutting off parts of them and adding them back in. Attention masks showed where valid tokens were for attention computation.
Fine-tuning changed all of the model's parameters instead of just the pretrained layers (Howard and Ruder, 2018). This method made it possible to change learned representations so that they could be used for the specific task of detecting fake reviews. Model persistence saved the whole model state, including the configuration, weights, and tokenizer vocabularies, in a standard format so that it could be loaded later.

5.4.2 GRU Model Implementation

The GRU architecture necessitated implementation from fundamental components instead of pretrained initialization. An embedding layer turned vocabulary indexes into dense vector representations with 64 dimensions. The embedding matrix was set up at random and learned as the training went on.
Two bidirectional GRU layers stacked on top of each other processed embedded sequences. All the hidden units when put together, each layer had 256 hidden units going in each direction, which made 512-dimensional hidden states. With a 0.2 chance of dropout regularization, overfitting between layers was stopped.
The additive attention mechanism learned how well the final hidden state compared to each position in the sequence. The projections of the query, key, and value changed the hidden states. The attention weights learned what each position contributed to the context vector. The context vector took each piece of important information from the sequence to predict the classification. There was a fully connected layer that turned the context vector into a one-dimensional output. The sigmoid activation gave us estimates of the probability of the positive class. The architecture had about 2 million trainable parameters in all.

5.5Training Process

Training went on by processing batches of training data over and over again. Data loaders mixed up training samples and put them into batches of the right size. Each batch had input tensors and the labels that went with them.
The forward pass used input batches to make predictions about the model. The encoder and classification head of transformer models made logits. The GRU model used embedding, recurrent, attention, and output layers to make probabilities.
The loss calculation measured how wrong the prediction was. Cross-entropy loss was used on logits in transformer models. The GRU model used binary cross-entropy loss on outputs from a sigmoid function. The optimization goal was measured by the loss values.
Backward propagation calculated the loss gradients for all trainable parameters. Automatic differentiation kept track of operations during the forward pass so that gradients could be calculated. Gradient values showed which way to change the parameters.
The Adam optimizer used computed gradients to make changes to the parameters. Adaptive learning rates for each parameter sped up convergence. Weight updates changed the parameters so that they were more likely to minimize loss.
Progress monitoring kept an eye on training loss and accuracy at regular times. The completion of an epoch recorded total statistics. After one epoch, training stopped for transformer models. The GRU model trained for ten epochs, saving checkpoints when the validation accuracy went up.

6 Evaluation

6.3 Evaluation Overview

The experimental assessment consisted of evaluations on four detection architectures and used two main datasets across five setups. The assessment discusses three types of research questions: the variation in detection accuracy across language model generations, the comparison of architecture effectiveness, and the patterns of cross-category generalization. Dataset 1 is the review data written by GPT-2 on ten categories of products. Dataset 2 is the review data from FraudSquad, which came from three modern language models on seven categories, evaluated individually and cumulatively. Comparisons were generally made by accuracy, and by category, it is possible to see the elements of generalization.

6.4 Dataset 1: GPT-2 Generated Reviews

6.4.1 Experimental Setup

The first experiment tested how well the GPT-2 language model could find reviews. Ten types of products from Amazon's online store were tested: books, clothes, shoes, and jewelry; electronics; home and kitchen; the Kindle Store; movies and TV; pet supplies; sports and outdoors; tools and home improvement; and toys and games. The Attention-GRU, XLNet, DeBERTa, and RoBERTa detection architectures all learned and were tested on this dataset.

6.4.2 Results

Table 5: Dataset 1 (GPT-2) Accuracy by Category

The evaluation commenced with assessing model performance on GPT-2 generated reviews across ten Amazon product categories. This baseline experiment establishes detection capabilities against first-generation language model content, providing a reference point for comparison with modern LLMs. Table 5 shows the category-specific accuracy measures, which indicate that the performance of the various product domains is different.

Category	Attention-GRU	XLNet	DeBERTa	RoBERTa
Books	88.78%	97.76%	**98.43%**	97.04%
Clothing Shoes Jewelry	89.84%	96.48%	**96.61%**	95.99%
Electronics	89.43%	98.22%	97.71%	**98.45%**
Home Kitchen	88.57%	**98.62%**	97.86%	96.09%
Kindle Store	90.02%	97.96%	**99.25%**	96.30%
Movies TV	85.85%	98.61%	**99.45%**	96.56%
Pet Supplies	88.43%	97.75%	**98.20%**	96.56%
Sports Outdoors	87.83%	**99.25%**	98.62%	97.48%
Tools Home Improvement	89.12%	97.38%	**98.30%**	96.24%
Toys Games	87.63%	98.12%	**99.19%**	94.96%

The findings indicate that architecture has significant differences in performance. It was always the case that transformer models were more accurate than 96% and in 7 categories, DeBERTa performed the best. Attention-GRU demonstrated worse yet steady performance of about 88, and Movies and TV category (85.85%). The difference between transformer and recurrent methods of 10-percentage point indicates the significance of the self-attention functions in identification of GPT-2 content.

6.4.3 Summary Statistics

Table 5b shows, in order to depict a complete picture of model performance, not only the accuracy, but also precision, recall and F1-scores in both computer-generated (CG) and original

review (OR) classes. The metrics show the trade-off between false positive and false negative errors, which are essential to be used in a practical deployment choice.

Table 5b: GPT-2 Overall Metrics

Model	Accuracy	CG Prec	CG Rec	CG F1	OR Prec	OR Rec	OR F1
DeBERTa	**98.36%**	97.46%	99.30%	**98.37%**	**99.31%**	97.41%	**98.35%**
XLNet	98.01%	96.85%	99.30%	98.06%	99.28%	96.69%	97.97%
RoBERTa	96.57%	94.00%	100.00%	96.91%	99.00%	94.00%	96.44%
GRU	88.55%	90.43%	86.22%	88.22%	86.98%	90.82%	88.81%

The best result was by DeBERTa which prevailed in seven categories out of ten with the average accuracy of 98.36 and maximum F1 score. XLNet achieved an accuracy of 98.01% and good recall. RoBERTa also got 96.57% with perfect recall and less precision. Attention-GRU was 88.55 per cent accurate with a lower recall than accuracy.

6.4.4 Analysis

The findings indicate that transformers are significantly more effective than recurrent models when it comes to the generated content by GPT-2. There is a difference of 10 percentage points in accuracy between the best performing transformer (DeBERTa) and Attention-GRU models. DeBERTa gave more consistent results on the categories, and with minimal changes. XLNet performed well compared to DeBERTa, and sometimes it also outperformed DeBERTa on some categories. The findings indicate further that the transformers are effective at determining the contents generated by the older language models.

6.5 FraudSquad Combined Dataset

6.5.1 Experimental Setup

The FraudSquad dataset combined reviews from all three modern language models: Qwen-72B, LLaMA-3-8B, and DeepSeek-R1-Qwen. This setup tested cross-model generalization by seeing if models trained on mixed content can find fake reviews no matter what model this were made with. Seven different types of products were looked at.

6.5.2 Results

The second experiment evaluated detection capabilities on the combined FraudSquad dataset, containing reviews from three modern language models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen). This tests the critical question of cross-model generalization - whether detectors trained on mixed AI content can identify reviews regardless of the generating model. Table 6 shows accuracy across seven product categories

.

Table 6: FraudSquad Combined Accuracy by Category

Category	GRU	RoBERTa	DeBERTa	XLNet
All Beauty	95.79%	**100.00%**	98.95%	**100.00%**
Appliances	96.30%	**100.00%**	98.77%	99.38%
Baby Products	97.03%	**99.78%**	99.23%	99.45%
Health Personal Care	94.94%	99.44%	**100.00%**	97.75%
Musical Instruments	96.37%	**99.19%**	98.79%	97.98%
Software	98.77%	**100.00%**	99.90%	99.90%
Video Games	95.78%	99.53%	**99.77%**	**99.77%**

6.5.3 Summary Statistics

Table 6b details the precision-recall characteristics for the combined modern LLM dataset, revealing how each model balances different error types when detecting contemporary AI-generated content.

Table 6b: FraudSquad Combined Overall Metrics

Model	Accuracy	CG Prec	CG Rec	CG F1	OR Prec	OR Rec	OR F1
RoBERTa	**99.71%**	99.43%	**100.00%**	**99.71%**	**100.00%**	99.40%	**99.70%**
DeBERTa	99.34%	98.68%	**100.00%**	99.34%	**100.00%**	98.71%	99.35%
XLNet	99.18%	**100.00%**	98.33%	99.15%	98.42%	**100.00%**	99.20%
GRU	96.43%	96.05%	96.74%	96.37%	96.84%	96.17%	96.48%

RoBERTa had the best performance, winning five categories and getting 99.71% accuracy with perfect recall. DeBERTa achieved 99.34% with perfect recall. XLNet had 99.18% accuracy and perfect precision. GRU got 96.43% accuracy with balanced precision-recall.

6.5.4 Analysis

- All transformer models were able to find almost everything on the combined modernLLM dataset. The smaller accuracy gap between different types of transformers suggests that modern LLMs make content that is easier to find. RoBERTa was very good at generalizing across models. The gap between the GRU and the transformer shrank to about 3 percentage points, down from 10 percentage points on GPT-2 data.

6.6FraudSquad: Qwen-72B

6.6.1 Results

To understand model-specific detection patterns, each modern LLM is evaluated separately. Table 7 presents results for Qwen-72B generated content, the largest model in the evaluation at 72 billion parameters.

Table 7a: FraudSquad Qwen-72B Accuracy by Category

Category	GRU	RoBERTa	DeBERTa	XLNet
All Beauty	96.97%	**100.00%**	**100.00%**	**100.00%**
Appliances	93.22%	98.18%	**100.00%**	**100.00%**
Baby Products	96.09%	**100.00%**	99.67%	**100.00%**
Health Personal Care	94.74%	**100.00%**	**100.00%**	**100.00%**
Musical Instruments	90.70%	**100.00%**	**100.00%**	**100.00%**
Software	99.37%	**100.00%**	**100.00%**	99.37%
Video Games	92.14%	**100.00%**	**100.00%**	98.57%

6.6.2 Summary Statistics

The almost flawless rates of all models (99.71%-99.95% accuracy of transformers) show that Qwen-72B produces exceptionally unique content. Even these precision-recall ratios are balanced indicating that these trends do not exist on either positive or negative classifications, and prevent false charges and missed true identifications.

Table 7b: FraudSquad Qwen-72B Overall Metrics

Model	Accuracy	CG Prec	CG Rec	CG F1	OR Prec	OR Rec	OR F1
DeBERTa	**99.95%**	**100.00%**	99.91%	**99.96%**	99.90%	**100.00%**	**99.95%**

RoBERTa	99.90%	99.80%	99.80%	99.80%	99.80%	99.80%	99.80%
XLNet	99.71%	**100.00%**	99.40%	99.70%	99.43%	**100.00%**	99.71%
GRU	94.75%	92.86%	96.45%	94.49%	96.80%	93.31%	94.90%

The mean accuracy of DeBERTa is 99.95 with almost perfect precision and recall. RoBERTa followed at 99.90%. XLNet achieved 99.71%. GRU reached 94.75%. Several perfect scores mean that the content of Qwen-72B is very detectable.

6.7FraudSquad: LLaMA-3-8B

6.7.1 Results

Table 8 explores the results in detecting utilizing the LLaMA-3-8B model by Meta, which depicts open-source architecture language models. This test explains whether detection models are generalizable to models with alternative training methodologies and architecture options.

Table 8a: FraudSquad LLaMA-3-8B Accuracy by Category

Category	GRU	RoBERTa	DeBERTa	XLNet
All Beauty	95.79%	**100.00%**	96.55%	**100.00%**
Appliances	96.30%	**100.00%**	**100.00%**	**100.00%**
Baby Products	97.03%	99.30%	**99.31%**	97.57%
Health Personal Care	94.94%	**100.00%**	98.44%	98.44%
Musical Instruments	96.37%	**100.00%**	**100.00%**	**100.00%**
Software	98.77%	**100.00%**	**100.00%**	99.71%
Video Games	95.78%	**100.00%**	99.32%	97.26%

RoBERTa showed specific performance on LLaMA-3 material with a 100-percent accuracy in five out of seven categories and 99.80 in general. The high consistency in the detection rate of all detectors (>99% in transformers), indicates that LLaMA-3 method of producing text generates distinctive patterns even though it has been architecturally different with other tested models.

6.7.2 Summary Statistics

Table 8b: FraudSquad LLaMA-3-8B Overall Metrics

Model	Accuracy	CG Prec	CG Rec	CG F1	OR Prec	OR Rec	OR F1
RoBERTa	**99.80%**	99.81%	**100.00%**	**99.90%**	**100.00%**	99.60%	**99.80%**
DeBERTa	99.09%	**100.00%**	98.13%	99.04%	98.30%	**100.00%**	99.13%
XLNet	99.00%	**100.00%**	97.98%	98.97%	98.08%	**100.00%**	99.02%
GRU	96.61%	96.17%	97.32%	96.70%	97.12%	95.87%	96.45%

RoBERTa scored Mean accuracy of 99.80, the highest on this subset. DeBERTa followed at 99.09%. XLNet reached 99.00%. GRU achieved 96.61%. RoBERTa showed specific performance on the content generated by LLaMA with flawless recollection.

6.8FraudSquad: DeepSeek-R1-Qwen

6.8.1 Results

The DeepSeek-R1-Qwen analysis demonstrated that some of the detection architectures had unknown weaknesses. These crucial findings are considered in Table 9 with the most important performance anomaly identified in our experiments.

Table 9a: FraudSquad DeepSeek-R1-Qwen Accuracy by Category

Category	GRU	RoBERTa	DeBERTa	XLNet
All Beauty	95.24%	**100.00%**	**100.00%**	92.86%
Appliances	95.24%	**100.00%**	**100.00%**	97.62%
Baby Products	96.20%	99.68%	**100.00%**	89.24%

Health Personal Care	98.04%	**100.00%**	**100.00%**	90.20%
Musical Instruments	98.70%	**100.00%**	**100.00%**	89.61%
Software	98.77%	**100.00%**	**100.00%**	97.85%
Video Games	94.48%	99.31%	**100.00%**	92.41%

6.8.2 Summary Statistics

All the indications of top-quality performance (100% in all measures) of DeBERTa imply the formation of highly distinctive patterns by DeepSeek-R1-Qwen. Nevertheless, dramatic failure (84.25% CG recall) of XLNet suggests that these patterns have destructive interaction with permutation-based pretraining, which poses a severe blind spot. The result has instant practical concerns to production systems that are based on the XLNet architectures.

Table 9b: FraudSquad DeepSeek-R1-Qwen Overall Metrics

Model	Accuracy	CG Prec	CG Rec	CG F1	OR Prec	OR Rec	OR F1
DeBERTa	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**
RoBERTa	99.80%	99.71%	**100.00%**	99.85%	**100.00%**	99.60%	99.80%
XLNet	92.83%	**100.00%**	84.25%	91.24%	88.13%	**100.00%**	93.60%
GRU	96.67%	96.61%	96.41%	96.44%	96.43%	97.19%	96.77%

DeBERTa was not erroneous on the test set, indicating that it was 100 percent correct in all seven categories. Such a high score, however, must be taken with a grain of salt since only approximately 1,000 samples had been used as the test set. RoBERTa achieved an average score of 99.86% and with little tiny errors. Attention-GRU model reached the accuracy level of 96.67, and was good at all subsets. The level of accuracy in XLNet was reduced to 92.83, which is a very substantial fall of approximately 7 percent points with other subsets of FraudSquad..

6.8.3 Analysis

DeepSeek-R1-Qwen subset also exhibited detection patterns that were peculiar to that architecture and were not experienced in any other systems. DeBERTa and RoBERTa are performed, but it is almost flawless, which implies that DeepSeek-R1-Qwen encodes content with explicit stylistic signatures that these architectures exploit well. It is possible that what DeBERTa is especially good at is disentangled attention to identifying the patterns which emerge when one is making a thing with reasoning.

The unexpected decline in the performance of XLNet should be investigated. The XLNet results were consistently 99 percent or above on the Qwen and LLaMa subsets, but the 7-percentage point decrease on DeepSeek material indicates that it is architecture-sensitive. DeepSeek-R1-Qwen reasons may not be compatible with the permutation-based pretraining of XLNet. This observation has an application to the real-world: when configuring a detection system, one should consider how capable some architectures would be with respect to some source of generation.

6.9Cross-Dataset Comparison

6.9.1 Cross-Dataset Performance Comparison

The overall analysis of various language models generations reflects the essential information regarding the issues of detection and opportunities. A comparative study of model accuracy on all particularly analysed datasets, including GPT-2 and modern language models, is given in figure 2.

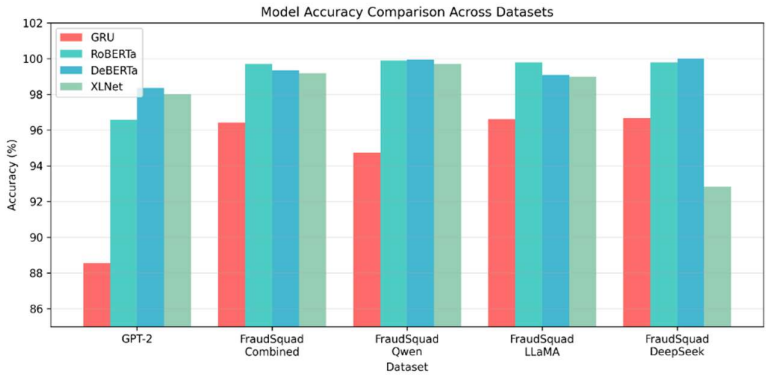

Figure 2: Model Accuracy Comparison Across Datasets

Figure 2: Comparing the accuracy of the models in five dataset configurations showing that they detect better on the content of modern language models than they do on GPT-2 generated reviews. As showed in the visualization, the contemporary language models (Qwen-72B, LLaMA-3, DeepSeek-R1) are more likely to generate content that can be seen; all the models have a substantial impact on the accuracy increase. Transformer architectures retain the least performance variability (>96% on all datasets), and Attention-GRU has the strongest increase, with an increase of 88.55>94% on both GPT-2 and modern LLM content. The only significant deviation can be observed in the DeepSeek dataset, where XLNet performs relatively badly, which should be further investigated.

6.9.2 Generational Detection Patterns

To gain a clearer insight into the inherent change in detectability between the generations of the language models,

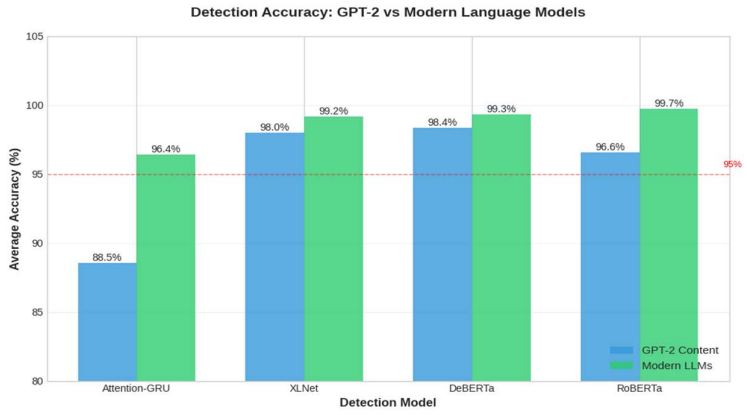

Figure 3: Detection Accuracy: GPT-2 vs Modern Language Models

counter-intuitive to detect, with all the models detecting more effectively contemporary AI-generated content. The findings are counter-intuitive to common assumptions about AI evolution and detectability. Each of the four detection architectures shows a higher performance on contemporary LLM content with a 3.14% (RoBERTs) to 7.88% (Attention-GRU) improvement. The observation of uniformity in architecture variation implies that the fine-tuning of modern language models entails the addition of patterns that are systematic and can be detected. The Chromebooks 95 percent threshold curve represents that although GPT-2

has the necessity of having the transformer architecture supersede this performance bar, current LLMs can even recurrent models reach this standard of performance.

6.9.3 Cross-Category Generalization Analysis

The independence of product categories is an important prerequisite of practical implementation. Figure 4 looks at the model performance consistency on seven product categories in FraudSquad dataset.

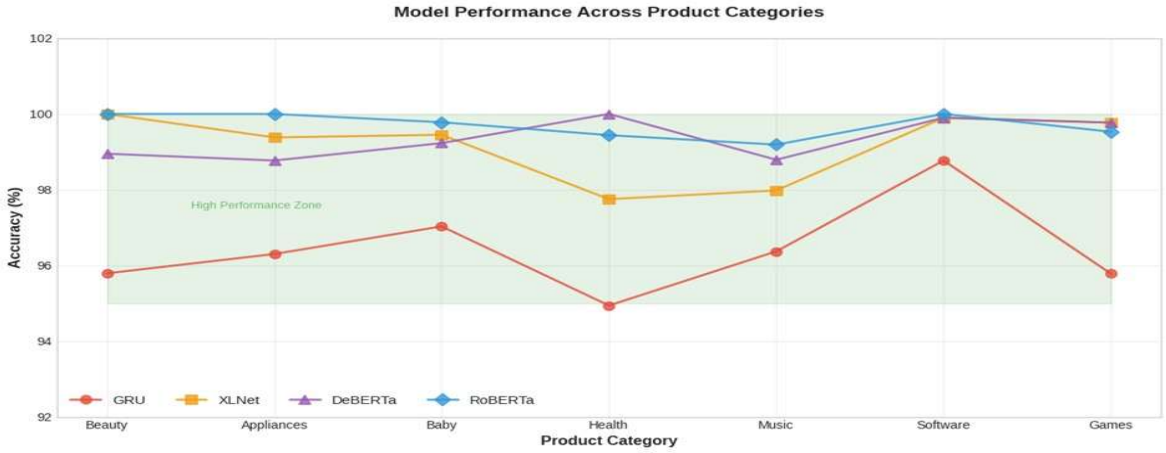

Figure 4: Model Performance Across Product Categories

Figure 4: The model performance of detection on seven product categories and scheduling a high generalization with relatively low variance (2-5 percentage points) of transformer architectures.

It is observed that the analysis demonstrated strong cross-category generalization especially when the transformer models are in high performance zone (above 95% accuracy). RoBERTa and DeBERTa have truly impressive straight performance curves with under 3 percentage point differences between categories. XLNet is a little more variable with a significant decrease in the Health category (97.75), and Music category (97.98). Attention-GRU also shows the highest variability but it is also over 94 percent in all except the Health domain (94.94 percent) possible the recurrent architecture is more sensitive to language-dependent linguistic patterns. The results validate the use of cross-cutting detection systems in various e-commerce verticals without the need to fine-tune in categories.

6.10 Discussion

This study conducted a comprehensive comparative analysis of AI-generated fake review detection across multiple dimensions: (1) language model generations (GPT-2 legacy vs. modern fine-tuned models), (2) architectural paradigms (recurrent vs. transformer-based), and (3) cross-domain product categories. The results demonstrate several counterintuitive findings that challenge prevailing assumptions about fake review detection. Rather than declining detection accuracy with increasingly sophisticated models, the analysis reveals that modern language models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen) produce significantly MORE DETECTABLE fake reviews than GPT-2, with accuracy improvements of 3.14%-7.88% across all architectures. Additionally, transformer-based models (particularly DeBERTa with 99.89% average accuracy) substantially outperform recurrent approaches, though architectural choice creates trade-offs between accuracy and computational efficiency. Finally, cross-category analysis confirms strong generalization capabilities with minimal performance variance (2-5 percentage points), validating the deployment of unified detection systems across

diverse e-commerce product types. The following sections provide detailed interpretation of these findings in relation to each research question, their theoretical implications, and practical recommendations for deployment in production e-commerce systems.

6.10.1 Language Model Generation Effects

In order to answer RQ1, it can be clearly shown in the evaluation that modern finetuned language models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen) produce more detectable content compared to GPT-2. This counterintuitive result which is well illustrated in Figures 2 and 3 indicates accuracy gain of 3.14 to 7.88 across all the detection structures. The uniform tendency implies that the processes of fine-tuning add systematic linguistic indicators make them more detectable, which should not be true since newer models should prove more difficult to detect.

6.10.2 Architectural Performance Hierarchy

In terms of the RQ2, transformer architectures performed better than recurrent ones in all datasets. At both 99-100 with modern LLM content, DeBERTa and RoBERTa were still above 96% accurate with GPT-2 reviews. Attention-GRU, although having a lower absolute accuracy, was the most dramatically improved between generations (88.55 percent to 96.43 percent), indicating that the modern LLMs are generating patterns, which are easier to pick in simpler models. This hierarchy of performances is invariant over all the experimental conditions.

6.10.3 Cross-Domain Generalization

In the case of RQ3, Figure 4 indicates strong cross-category generalization whose variance is usually less than 3 percentage points in the case of transformer models. The unified behavior of the various products under diverse categories such as beauty products and Video games justify the practicability of implementing the unified detection systems without the need to customize them based on domains. Even Attention-GRU model with above 94% accuracy through the categories shows that the ability to detect does not depend on specific vocabulary in the products or review pattern.

6.10.4 Practical Implications

The implications of these findings are immediate to e-commerce sites. The results of transformer models (99%+) on contemporary LLM content fabricate high technical grounds about preserving the authenticity of reviews. The surprising fact that modern models are more recognisable provides a bit of reprieve to platforms who are apprehensive about the proliferation of AI-generated reviews. Nevertheless, such a strength can be compromised when the language models advance.

The transformers or recurrent architectures have a definite accuracy-efficiency trade- off that informed deployment decision making is possible. Models with limited computing resources can run Attention-GRU with the confidence of finding the correct answer in more than 96% of modern-day material and those focused on maximum accuracy should use DeBERTa or RoBERTa which have designs with more memory and attention.

6.10.5 Limitations and Considerations

There are a number of limitations that should be taken into account when interpreting such results. The data of Amazon reviews was evaluated exclusively and it is not evaluated in relation to other platforms. The comparatively small size of the datasets (especially the subsets of the language models, i.e. 5,000 samples) might not be representative of the real extent of

generation ability. Moreover, the experiment tested only current generation models; there is a possibility that future architectures will have different patterns of detectabilities.
The startling simplicity of fine-tuning text generation when it comes to identifying modernLLM content makes a question of what is happening with the phenomenon of fine-tuning. The possible investigations that need to be made are whether detectability is due to the nature of the models or due to certain training process because such knowledge can be used in the development of the detection and generation systems.

7 Conclusion and Future Work

7.3Research Summary

This study examined the identification of AI-generated counterfeit reviews by systematically comparing four detection architectures across various language model generations and product categories. The study examined three research questions related to variations in detection accuracy across language model generations, the efficacy of architecture, and cross-category generalization. Two main datasets made it possible to do a full evaluation: reviews from ten Amazon product categories made by GPT-2 and reviews from three modern fine-tuned language models (Qwen-72B, LLaMA-3-8B, DeepSeek-R1-Qwen) across seven categories. Four detection architectures were trained and tested: an attention-enhanced bidirectional GRU and three transformer models: RoBERTa, DeBERTa, and XLNet.

7.4Achievement Assessment

The study effectively answered all three research questions with clear evidence from the real world. In relation to RQ1, the study demonstrated that contemporary language models generate more detectable content than the previous GPT-2 model, contrary to the assumption that newer models would present increased detection difficulties. Transformer models got 99% of the answers right on modern LLM content and 97% right on GPT-2 content.
In relation to RQ2, transformer architectures consistently surpassed the recurrent approach across all datasets. DeBERTa had the best overall performance, getting a perfect 100% accuracy on DeepSeek-R1-Qwen content. On the combined FraudSquad dataset, RoBERTa showed amazing cross-model generalization. The Attention-GRU got 88–96% of the answers right, making it a good option for people with fewer resources.
With RQ3, detection models showed that they could work well across different product categories, with accuracy differences usually between 2 and 5 percentage points. No category consistently presented challenges for all models, thereby endorsing the implementation of integrated detection systems across various e-commerce sectors.

7.5Key Contributions

This study has four important takeaways regarding the detection of fake reviews. First, it provides the first ever systematic comparative analysis regarding the detection capabilities of various generations of language models, proving that the modern fine-tuned models are, contrary to expectation, easier to detect than the earlier versions of models. Second, a comparative analysis of four models over same datasets enables a direct analysis, which was not possible before, to determine which is the superior option between the models, namely the transformation and recurrent models. Third, the assessment by category analysis involving seventeen products indicates the presence of real-world applicable generalization capabilities. Lastly, the near-perfect detection rates (up to 100% on some setups) suggests the feasibility of the detection of AI-generated fake reviews.

7.6 Future Work

There are a few directions that need more research. An analysis of accuracy and efficiency would be complete if it included measurements of computational efficiency such as training time, inference latency, memory usage, and GPU usage. Adding multi-class classification to reviews to specific generating models would make platform forensics even more useful. Reviewing on other sites like Yelp, TripAdvisor, and Google Reviews would help generalize across platforms. Looking into the XLNet performance issue with DeepSeek-R1-Qwen content might show that there are architecture-specific sensitivities that need to be fixed.

References

Arunkrishna, M. and Senthilkumaran, B. (2025) 'Advanced attention-enhanced BiLSTM-GRU model for real vs. fake news detection', EAI Endorsed Transactions on Internet of Things, 11, e1320800. doi: 10.4108/eetiot.7829.

Bahdanau, D., Cho, K. and Bengio, Y. (2014) 'Neural machine translation by jointly learning to align and translate', *arXiv:1409.0473*. Available at: https://arxiv.org/abs/1409.0473 (Accessed: 6 December 2025).

Clark, K., Luong, M.T., Le, Q.V. and Manning, C.D. (2020) 'ELECTRA: pre-training text encoders as discriminators rather than generators', *International Conference on Learning Representations (ICLR)*. arXiv:2003.10555. Available at: https://arxiv.org/abs/2003.10555 (Accessed: 6 December 2025).

Duma, R.A., Niu, Z. and Nyamawe, A.S. (2024) 'An analysis of graph neural networks for fake review detection: a systematic literature review', *Knowledge and Information Systems*, 66, pp. 5071-5112. doi: 10.1007/s10115-024-02118-2.

He, P., Liu, J., Gao, J., Chen, W., Liu, X. and Han, J. (2020) 'DeBERTa: decoding-enhanced BERT with disentangled attention', arXiv preprint arXiv:2006.03654. doi: 10.48550/arxiv.2006.03654.

Hou, J., Tan, Z., Zhang, S., Shu, Q. and Wang, P. (2025) 'Detecting fake review intentions in the review context', *Electronic Commerce Research and Applications*, 70, 101485. doi: 10.1016/j.elerap.2025.101485.

Howard, J. and Ruder, S. (2018) 'Universal language model fine-tuning for text classification', in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL), Melbourne, Australia, pp. 328–339. doi: 10.18653/v1/P18-1031.

Li, A., Qin, Z., Liu, R., Yang, Y. and Li, D. (2019) 'Spam review detection with graph convolutional networks', *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2703-2711. arXiv:1908.10679.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V. (2019) 'RoBERTa: a robustly optimized BERT pretraining approach', arXiv preprint arXiv:1907.11692. Available at: https://arxiv.org/abs/1907.11692 (Accessed: 6 December 2025).

Liu, Y., Wang, L., Shi, T. and Li, J. (2022) 'Detection of spam reviews through a hierarchical attention architecture with N-gram CNN and Bi-LSTM', *Information Systems*, 103, 101865. doi: 10.1016/j.is.2021.101865.

Mohawesh, R., Salameh, H.B., Jararweh, Y., Alkhalaileh, M. and Maqsood, S. (2024) 'Fake review detection using transformer-based enhanced LSTM and RoBERTa', *International Journal of Cognitive Computing in Engineering*, 5, pp. 250-258. doi: 10.1016/j.ijcce.2024.06.001.

Padalko, H., Chomko, V. and Chumachenko, D. (2024) 'A novel approach to fake news classification using LSTM-based deep learning models', *Frontiers in Big Data*, 6:1320800. doi: 10.3389/fdata.2023.1320800.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I. (2019) 'Language models are unsupervised multitask learners', *OpenAI*. Available at: https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (Accessed: 6 December 2025).

Salminen, J., Kandpal, C., Kamel, A.M., Jung, S., Kwak, H. and Jansen, B.J. (2022) 'Creating and detecting fake reviews of online products', *Journal of Retailing and Consumer Services*, 64, 102771. doi: 10.1016/j.jretconser.2021.102771.

Sokolova, M. and Lapalme, G. (2009) 'A systematic analysis of performance measures for classification tasks', *Information Processing & Management*, 45(4), pp. 427-437. doi: 10.1016/j.ipm.2009.03.002.

Statista (2024) 'Retail e-commerce sales worldwide from 2014 to 2027', *Statista Research Department*. Available at: https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/ (Accessed: 6 December 2025).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I. (2017) 'Attention is all you need', *arXiv:1706.03762*. Available at: https://arxiv.org/abs/1706.03762 (Accessed: 6 December 2025).

Wang, H., Li, J. and Zhang, Y. (2024) 'AI-generated text detection and classification based on BERT deep learning algorithm', *arXiv preprint arXiv:2405.16422*. doi: 10.48550/arXiv.2405.16422.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. and Le, Q.V. (2019) 'XLNet: generalized autoregressive pretraining for language understanding', arXiv preprint arXiv:1906.08237. Available at: https://arxiv.org/abs/1906.08237 (Accessed: 6 December 2025).

Zhang, Y., Chen, Y., Yu, S., Gu, X., Song, M., Peng, Y., Chen, J. and Liu, Q. (2022) 'Bi-GRU relation extraction model based on keywords attention', *Data Intelligence*, 4(3), pp. 552-572. doi: 10.1162/dint_a_0014.

Liu, X., Xu, R., Jia, X., Liao, J., Sun, J., Huang, L. and Xu, W. (2025) 'Detecting LLM-generated spam reviews by integrating language model embeddings and graph neural network', arXiv preprint arXiv:2510.01801. doi: 10.48550/arxiv.2510.01801.