IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

When AI reads between the lines: Detecting hidden insincerity in text

Ramla Basharat[1], Zahoor-ur-Rehman[1]*, Tahira Amin[1], Fatema Sabeen Shaikh[2], Abdullah Alqahtani[2]

[1]*Department of Computer Science, COMSATS University Islamabad, Attock Campus, Pakistan.*
[2]*Computer Information Systems Department, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, KSA.*

Corresponding author: Zahoor-ur-Rehman (*Email: xahoor@cuiatk.edu.pk*)

Abstract

In the current digital age, technology has become an integral part of every facet of life, often shaping human perceptions through the content shared on social media platforms. However, this digital transformation has also led to the emergence of significant challenges, particularly the proliferation of irrelevant and harmful content, such as "toxic" material. This paper addresses a key issue faced by many online platforms, particularly in the context of Quora, which is the prevalence of short, insincere questions that lack meaningful content. These insincere questions are typically characterized by exaggerated falsehoods, argumentative tones, and unethical language, contributing to a negative user experience. This study focuses on tackling these challenges, specifically within the framework of a Kaggle competition, by utilizing innovative techniques to detect and mitigate problematic content. We propose a solution that leverages a pre-trained deep bidirectional model, specifically fine-tuned with BERT-base, to classify textual questions as either sincere or insincere. The model is built upon a series of carefully engineered features, applied after basic data preprocessing. Our proposed approach enables the automated identification of toxic content, allowing for the removal of insincere questions and thus enhancing the overall quality of information on the platform. The effectiveness of our model is demonstrated through its superior performance compared to existing classification methods, achieving an impressive F1 score of 0.721. This result highlights the potential of our approach in addressing the challenges posed by insincere and toxic content in online discussions.

Keywords: BERT-base, Bidirectional transformer, Insincerity, Social Platforms, Transformer based model.

DOI: 10.53894/ijirss.v8i6.10105
Funding: This study received no specific financial support.
History: Received: 4 July 2025 / **Revised:** 11 August 2025 / **Accepted:** 12 August 2025 / **Published:** 19 September 2025
Copyright: © 2025 by the authors. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).
Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

In this era of rapid digital communication, online platforms have become essential spaces for information exchange, debate, and discussion. Millions of users engage daily on forums like Quora, Reddit, and Stack Exchange, seeking answers, sharing insights, and shaping public discourse. However, not all interactions stem from a place of genuine curiosity. A growing concern in digital spaces is insincerity, a subtle yet pervasive issue where users craft misleading, manipulative, or provocative questions that distort discussions rather than contribute meaningfully. Insincerity, in the context of user-generated questions, refers to the deliberate framing of inquiries in a way that misleads, provokes, or serves an ulterior motive. Unlike outright misinformation, which involves falsehoods presented as facts, insincere questions often operate in the shadows of plausibility. They may be subtly manipulative, emotionally charged, or framed to elicit a biased response. For example, a politically motivated question might disguise an agenda under the pretense of neutral inquiry, while a technically misleading question could spread confusion in a professional discussion.

The impact of such content is far from trivial. A 2023 Pew Research Center report found that nearly 40% of internet users had encountered content they perceived as deliberately misleading or inflammatory. Similarly, a 2024 study by Redline Digital reported that 38.2% of U.S. news consumers had unknowingly shared misinformation on social media, further highlighting how deceptive content, whether insincere or outright false, shapes online narratives.

While extensive research exists on fake news detection and misinformation spread [1] the issue of insincerity, particularly in user generated questions, remains an underexplored domain. Sentiment analysis studies Umarani, et al. [2] can detect sarcasm or negativity in textual data, but these techniques fail to capture the subtleties of insincere questioning. Unlike blatant misinformation, insincere questions often exist in a gray area: they may not contain outright falsehoods but are deliberately framed to manipulate, mislead, or provoke. A key limitation in previous research is the heavy reliance on structured, formal datasets to train language models [3] While effective for controlled environments, these models struggle when faced with the unpredictable, chaotic nature of real-world online discourse.

User generated text is messy—full of abbreviations, colloquialisms, unconventional grammar, and creative spellings. This "freeform text" complicates natural language processing (NLP) tasks, as traditional models often falter when analyzing informal, unstructured content. Even more challenging is the contextual nature of insincerity. The same phrase might be harmless in one setting but manipulative in another. A politically charged question, for example, may carry an entirely different intent compared to a technical inquiry. Due to this fluidity, existing misinformation detection techniques, designed for more structured content, often fall short when applied to user generated forums.

Table 1.
Notations Used Within the Paper.

Annotation	**Expanded Form**
BERT	Bidirectional Encoder Representations from Transformers
BiCuDNNGRU	Bidirectional CuDNN accelerated Gated Recurrent Unit
BiCuDNNLSTM	Bidirectional CuDNN accelerated Long Short Term Memory
CapsNet	Capsule Network

This gap in the literature underscores the necessity of a specialized approach to detecting insincerity in online discussions. Existing frameworks primarily target fake content but fail to address the more nuanced problem of manipulative questioning. To bridge this gap, this study proposes a machine learning based framework for identifying insincere user generated questions. By leveraging linguistic cues, contextual irregularities, and manipulative phrasing patterns, our model aims to distinguish between sincere inquiries and those designed to mislead or provoke.

The contributions of this paper include:

- A novel framework that integrates multiple indicators of insincerity—contextual anomalies, linguistic markers, and semantic inconsistencies—to enhance insincerity detection.
- A multidimensional machine learning model designed to process freeform text with improved accuracy, overcoming the limitations of traditional structured data approaches.
- A comprehensive evaluation across diverse online platforms, demonstrating the model's adaptability, effectiveness, and scalability in real world settings.

By refining insincerity detection, this research contributes to the broader effort of ensuring authenticity and integrity in online discussions, ultimately fostering more meaningful and trustworthy digital interactions. Table I presents a comprehensive list of all the notations used throughout this study, providing clarity on the terms and symbols employed in our analysis.

The layout of this study is as follows: II reviews related work on misinformation detection, sentiment analysis, and insincerity classification, pointing out key challenges. III explains the proposed methodology, including dataset selection, feature extraction, and model development. IV portrays the experimental setup and V presents the results, comparing the model's performance with existing methods.

2. Literature Review

Detecting insincerity in textual data is closely linked to research in misinformation detection, sentiment analysis, and text classification. Over the years, numerous approaches have been explored, ranging from traditional supervised learning to advanced deep learning techniques. Survey studies such as Aïmeur, et al. [4]; Yuan, et al. [5] and Zhou and Zafarani [6] provide comprehensive overviews of misin- formation detection methods, highlighting the evolution of NLP techniques in handling deceptive content. These works emphasize the growing complexity of insincerity detection, especially in informal, user generated content where linguistic cues are often subtle and context dependent. Building on these foundations, this section categorizes existing approaches into two main areas: supervised learning techniques and deep learning based techniques for insincerity detection.

2.1. Supervised Learning Techniques

Supervised learning has been a dominant approach in classification tasks for insincerity detection, leveraging various machine learning models and linguistic feature sets. Several studies have demonstrated its effectiveness in different domains. Rubin, et al. [7] applied Support Vector Machine (SVM) on a dataset of 360 news articles, incorporating linguistic and stylistic features such as grammar, punctuation, humor, negative affect, and absurdity. Their model achieved an F1-score of 87%, with absurdity, grammar, and punctuation proving to be the most influential features. Reyes and Rosso [8] explored irony detection using Naïve Bayes, Decision Trees, and SVM, applying 10-fold cross-validation on a dataset of ironic reviews. Their approach identified six key linguistic characteristics, including n-grams, part-of-speech profiling, and affective profiling, to enhance classification accuracy.

In another study, Chen, et al. [9] introduced a lexical syntactic feature based (LSF) model to detect and block offensive content in online comments. Their model, tested on YouTube comment boards, demonstrated SVM's superior performance over Naïve Bayes in identifying unethical content while accommodating informal and misspelled text. Van Hee, et al. [10] addressed the class imbalance problem in insincerity detection by using cost sensitive SVM, training their model on English and Dutch corpora from ASKfm. They reported an F1-score of 64.32% for English and 58.72% for Dutch, suggesting the approach's adaptability to different languages. Further expanding on supervised approaches, Lau, et al. [11] designed an SVM-based text mining model to detect semantic similarity in Amazon reviews. Their approach, supported by manual annotation, achieved a 95% true positive rate, revealing an estimated spam rate of only 2%. Nobata, et al. [12] demonstrated the effectiveness of a supervised classification model incorporating syntactic, embedding, and standard NLP features, outperforming deep learning models with an F1-score of 0.81. Other notable contributions include a multi-classifier fusion method for insult detection in Kaggle datasets [13] a chained classification approach for identifying sexual predators [14] and a logistic regression model for topic based tweet classification [15] which achieved a 92% accuracy rate.

Supervised learning has also been explored in multilingual and domain specific applications. Unsvåg and Gambäck [16] analyzed user characteristics on Twitter across English, Portuguese, and German datasets, though they found minimal correlation between user traits and hate speech. Al-Khatib and El-Beltagy [17] focused on Arabic emotion detection, utilizing a Twitter-based dataset to test Complement Naïve Bayes, achieving an accuracy of 68.1%. Ravi and Ravi [18] studied irony and satire detection in news, combining multiple feature extraction techniques and demonstrating that logistic regression and random forests performed well in ensemble settings. These studies highlight the continued reliance on supervised learning techniques for insincerity detection and their adaptability across various textual domains.

Chittari, et al. [19] developed a text classification model to detect insincere questions. They initially trained classical models like Logistic Regression and SVM as baselines before implementing neural network based approaches. Their best performing model, a two layer Bidirectional LSTM with word embeddings, achieved great results. Kumar, et al. [20] analyzed toxic content on social media using the Quora dataset to identify inappropriate questions that degrade platform quality. They applied tokenization, vectorization, and machine learning models like Naïve Bayes, Logistic Regression, SVM, and Random Forest. Their results showed that SVM outperformed other models. Nguyen and Meesad [21] analyzed the Quora dataset from Kaggle to detect insincere and spam content using preprocessing algorithms and PySpark based models. They also examined user writing patterns to enhance prediction accuracy. Their results showed that the Gradient Boosted Tree model performed best. Omarova, et al. [22] explored machine learning techniques, including Logistic Regression, Random Forest, and XGBoost, to identify semantically similar questions using the Quora Question Pairs dataset [23] introduced a multimodal dataset for detecting satire and sarcasm, built on the MUStARD dataset, incorporating both textual and visual cues. Their study experimented with models like LSTM, SVM, and Logistic Regression to enhance sarcasm and satire detection, aiming to improve understanding of humor and irony in online communication.

In Lappas [24] a real dataset was used to detect fake reviews by evaluating authenticity factors and attack strategies. Similarly, Malbon [25] highlighted how companies generate fake reviews to influence customers and suggested an alliance based approach for better regulation. A collective classification algorithm (MHCC) was introduced in Li, et al. [26] and later extended to a PU Learning model, improving detection by iteratively identifying fake reviews from unlabeled data. For abuse detection, Papegnies, et al. [27] proposed a graph based system using SVM, outperforming content based methods. Meanwhile, Waseem and Hovy [28] provided a hate speech dataset and found that adding gender information significantly improved classification performance.

In Mungekar, et al. [29] the Quora Insincere Question classification was performed using Naïve Bayes, Logistic Regression, SVM, Decision Tree, and Random Forest, with Decision Tree showing the best results. Gaire, et al. [30] used RNN to address the problem, implementing algorithms like Multinomial Naive Bayes, k-nearest, and Logistic Regression. In Mediratta and Oswal [31] various models like SVM, Naïve Bayes, GRU, and LSTM were employed, with RandomUnderSampler improving results, especially when using GRU with GloVe embeddings. Ranganathan, et al. [32] applied Stochastic Gradient Descent with SVM and introduced a keyword identification layer, achieving 47.52% accuracy. Mujadia, et al. [33] focused on binary classification of Quora questions and used gradient boosting, random forest, and k-nearest neighbors, with Adaboost performing best with TF-IDF features. Al-Ramahi and Alsmadi [34] compared bag-of-words and n-gram representations with Term Frequency (TF) and Term Presence (TP), finding the latter to be more effective, with Logistic Regression yielding improved results.

2.2. Deep Learning Techniques

Various Neural Network algorithms, particularly Recurrent Neural Networks (RNNs), have shown promising results in textual data analysis, especially for tasks involving sequential data manipulation like time series prediction, speech recognition, and text generation. However, RNNs face the significant issue of vanishing gradients, where the error decays severely as it propagates through layers, effectively preventing the network from learning. This challenge was highlighted by Kshirsagar, et al. [35] who presented a neural classification system for detecting hate speech using pre-processed text and modified word embed- dings on three publicly available datasets.

In cyberbullying detection, Agrawal and Awekar [36] employed deep learning based models using three datasets to classify cyberbullying across multiple platforms, showing that these models can capture isolated features for improved detection. Similarly, Lu, et al. [37] introduced the Char-CNNS model for detecting textual cyberbullying, focusing on character level features and addressing spelling errors using shortcuts in the Chinese Weibo dataset. They overcame class imbalance by using a focal loss function, achieving better results on the Weibo dataset compared to other models like SVM, linear regression, and CNN. Further, Bu and Cho [38] proposed a hybrid model combining character level CNN and word level LRCN in TensorFlow for cyberbullying detection, obtaining superior results using word embedding techniques and knowledge transfer methods. For sarcasm detection, Kumar and Garg [39] utilized the SemEval, Twitter streams, and Reddit datasets with various machine learning techniques, including KNN, SVM, decision trees, random forests, and neural networks. The study achieved a 92% accuracy on the Twitter dataset, using TF-IDF, statistical, and lexical features along with sentiment scores. Similarly, Cheong, et al. [40] targeted predatory behavior in online game chats, using sentiment and rule breaking features with classifiers like Naïve Bayes, Logistic Regression, and MLP, achieving a 92.51% accuracy and high F1 scores. Monti, et al. [41] proposed a propagation based approach for fake news detection, leveraging geometric deep learning to analyze social network structures and news dissemination patterns.

For inappropriate content detection, Yenala, et al. [42] proposed the Convolutional BiDirectional LSTM (CBiLSTM) architecture, combining CNNs and BiLSTMs for better query completion suggestion and user conversation filtering. The CBiLSTM model outperformed the BiLSTM model, achieving an F1 score of 0.8720 in automatic inappropriate query detection and conversation filtering tasks. Gottipati, et al. [43] explores neural network- based models for identifying insincere content, incorporating profanity features to enhance detection. Rachha and Vanmane [44] addressed the issue of insincere and toxic content on platforms like Quora, Reddit, and Twitter by fine tuning transformer based models such as BERT, RoBERTa, DistilBERT, and ALBERT for insincere question classification. Their study highlights the effectiveness of transfer learning in NLP for improving content moderation. Aslam, et al. [45] investigated insincere question classification on Quora using machine learning and deep learning models, including SVM, logistic regression, and LSTM. Their study, conducted with Kaggle data and MATLAB, demonstrated that LSTM outperforms traditional models in handling imbalanced datasets, highlighting its effectiveness for text classification in online forums. Chakraborty explored the detection of insincere content on Q&A platforms like Quora and Stack Overflow using transformer based models, including BERT, XLNet, StructBERT, and DeBERTa. Their findings demonstrated that even with limited computational resources, DeBERTa achieved the highest accuracy and F1-scores.

Ravi and Ravi [18] explored the use of deep learning and word embeddings for text classification, leveraging convolutional neural networks (CNNs) for improved accuracy. They proposed a convolution based truth discovery algorithm with multitasking to assess the genuineness of textual data. Arora, et al. [46] employed BERT-based frameworks to classify Quora questions as sincere or insincere, emphasizing the role of automated detection in maintaining constructive online interactions. Their approach, involving data preprocessing, finetuning, and performance evaluation, achieved a remarkable F1-score.

In Zhou, et al. [47] a crowdsourced knowledge graph was proposed to enhance fake news detection beyond linguistic features. Using the RSTVSM method, Rubin, et al. [48] clustered news articles Num_Capital_Char based on rhetorical relations, improving deception detection. A psycholinguistic based approach in Del Pilar Salas-Zárate, et al. [49] trained on the LIWC lexicon effectively classified satirical news on Spanish and Mexican Twitter datasets, achieving F-measures above 84%.

Mishra and Kumar [50] experimented with different variations of BERT for Quora Insincere Question classification, including BERT with CNN and a linear layer. The simple BERT model achieved a slight improvement in F1-score over the combined LSTM and GRU model, highlighting its effectiveness on the Quora dataset.

Table 2.
Comparative Analysis of Various Works, Methodologies, and Features.

Author	Method	Features
Chen, et al. [9]	LSF + SVM	Style Features, Structure Features, Cyberbullying Features
Van Hee, et al. [10]	SVM with linear kernel	Word n-gram bag-of-words, character n-gram bag-of-words
Lau, et al. [11]	SVM	Syntactical, lexical and stylistic features
Ben Ismail and Bchir [13]	Local multi-classifier fusion method	TF-IDF
Al-Khatib and El-Beltagy [17]	Naïve Bayes classifier	Bag-of-words
Ravi and Ravi [18]	Logistic Regression and Random Forest	Unigram, Semantic, Psycholinguistic and Statistical features
Agrawal and Awekar [36]	Neural Network Based Model	Bag-of-character n-gram, GloVe embeddings, SSWE embeddings
Yenala, et al. [42]	Convolutional, BiDirectional LSTM	Word embedding features
Bu and Cho [38]	Character-level CNN and word-level LRCN	Character-level and word-level features
Gaire, et al. [30]	RNN	GloVe and paragram word embeddings
Mediratta and Oswal [31]	GRU	GloVe word embedding
Mishra and Kumar [50]	BERT	BERT embeddings

3. Methodology

This study explores the application of deep learning models for the task of insincerity detection in user generated text, specifically focusing on questions posted on the Quora platform. The proposed methodology comprises three main stages: handcrafted feature extraction, preprocessing of the text data, and classification using four different neural architectures, Bidirectional CuDNNGRU, Bidirectional CuDNNLSTM, Capsule Network, and BERT Base. These models were selected based on their proven ability to model sequential and contextual relationships in natural language.

3.1. Feature Engineering

To help the models understand more than just the word meanings captured by embeddings, we created a set of extra features that describe the style and structure of each question. These handcrafted features focus on how the text is written and give the model more information that can help tell apart sincere and insincere questions.

First, we included lexical features that describe the wording and length of each question. One such feature is the average word length, which tells us how long the words are on average in a sentence; longer words might suggest more complex or formal language. We also measured the total number of words in a question, which can reflect whether a sentence is unusually short or overly long. To bring these two features together, we calculated a simple metric called word density, defined as the average word length divided by the total number of words as described by Equation 1.

$$\text{Word\_Density} = \frac{\text{Avg\_Word\_Length}}{\text{Total\_Words}} \tag{1}$$

This word density score gives us a sense of how tightly or loosely the sentence is written. It can sometimes highlight writing that feels exaggerated, overly dramatic, or unusually simple, all of which are often signs of insincere or misleading content.

Along with these lexical features, we also looked at the way characters are used in a sentence. One such feature is the number of capital letters in each question. In online text, capital letters are often used for emphasis or to express strong emotions like anger or sarcasm, things that may suggest insincerity. We also counted the total number of characters in the question. Using these two values, we calculated the capitalization ratio, which tells us what portion of the sentence is written in capital letters as seen through Equation 2.

$$\text{Capital\_Ratio} = \frac{\text{Num\_Capital\_Char}}{\text{Total\_Char}} \tag{2}$$

This ratio gives a sense of how noticeable or dominant capital letters are in the sentence, which can be a useful clue when deciding if the text sounds sincere or not. When combined with the earlier features, it helps enrich the dataset with important structural and stylistic details, things that might be missed by embeddings or attention layers on their own.

3.2. Preprocessing

The preprocessing stage was essential to prepare the textual data for model training and ensure consistency across all input samples. As a first step, all questions were converted to lowercase using a simple text transformation. This helps prevent the model from treating the same word differently just because of its case, such as interpreting "Sincere" and "sincere" as unrelated tokens.

Next, punctuation marks were removed from each question. Since punctuation generally does not contribute significant meaning in short question style texts, its removal helps reduce noise and simplify the input structure. This step was applied using standard text cleaning routines that filter out common punctuation characters.

Once the text was cleaned, it was tokenized into smaller units for model input. For the BERT-based architecture, Word Piece tokenization was applied to maintain compatibility with BERT's pretrained vocabulary. For the BiGRU, BiLSTM, and CapsNet models, standard word level tokenization was used to split the sentences into individual tokens.

These preprocessing steps resulted in clean, standardized inputs, allowing the models to better identify meaningful patterns in the data without being distracted by irregularities or noise in the raw text.

3.3. Model Architectures

To assess the effectiveness of different deep learning strategies in detecting insincere content, we implemented and evaluated four architectures: Bidirectional CuDNNGRU, Bidirectional CuDNNLSTM, Capsule Network, and the pretrained BERT Base model. Each model was carefully designed to capture distinct types of information from the text, ranging from sequential dependencies to hierarchical structures and contextual embeddings.

- Bidirectional CuDNNGRU: The Bidirectional CuDNNGRU model is built upon the Gated Recurrent Unit (GRU), a variant of RNN that balances expressive capacity and computational efficiency. For the task of insincerity detection, the GRU's ability to selectively retain and forget information across a sequence is particularly useful. Insincere questions often contain specific linguistic cues—such as sarcasm, exaggeration, or emotional emphasis, that may be scattered across different positions in the sentence. Capturing such patterns requires a memory mechanism capable of modeling contextual dependencies over time.

At each time step *t*, the GRU updates its hidden state using the reset gate r_t and update gate z_t, which control how much of the previous state is retained or replaced. The GRU computations are given by Equation 3:

$$\begin{aligned} z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\ \tilde{h}_t &= \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \end{aligned} \qquad (3)$$

Here, x_t represents the embedding of the *t*-th token, and h_{t-1} is the previous hidden state. By modulating how much prior context is retained, the GRU is able to emphasize important segments of the question that may signal insincerity.

To enhance contextual understanding, a Bidirectional GRU processes the question in both forward and backward directions. This allows the model to incorporate clues that appear later in the sentence, such as rhetorical tones or suggestive keywords, which may modify the interpretation of earlier content. The forward and backward hidden states are concatenated at each time step using Equation 4:

$$h_t^{\text{bi}} = [\overrightarrow{h_t}; \overleftarrow{h_t}] \qquad (4)$$

The sequence of h_t^{bi} vectors are passed through a Global MaxPooling1D layer, which captures the most activated features across time. This is especially effective for identifying sharp linguistic spikes, like all caps, emotionally loaded words, or abrupt transitions, that commonly appear in insincere posts. The pooled vector is then passed through a dense layer with ReLU activation to capture nonlinear feature interactions. A dropout layer (rate = 0.1) is applied to reduce overfitting. Finally, a sigmoid activated dense layer computes the binary probability $\hat{y}$ of insincerity using Equation 5:

$$\hat{y} = \sigma(W^T h + b) \qquad (5)$$

where *h* is the pooled representation, and *W* and *b* are the weights and bias of the final layer. The sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$ ensures the output lies in the range [0, 1], suitable for binary classification. The architecture of the proposed Bidirectional CuDNNGRU model is illustrated in Figure 1.

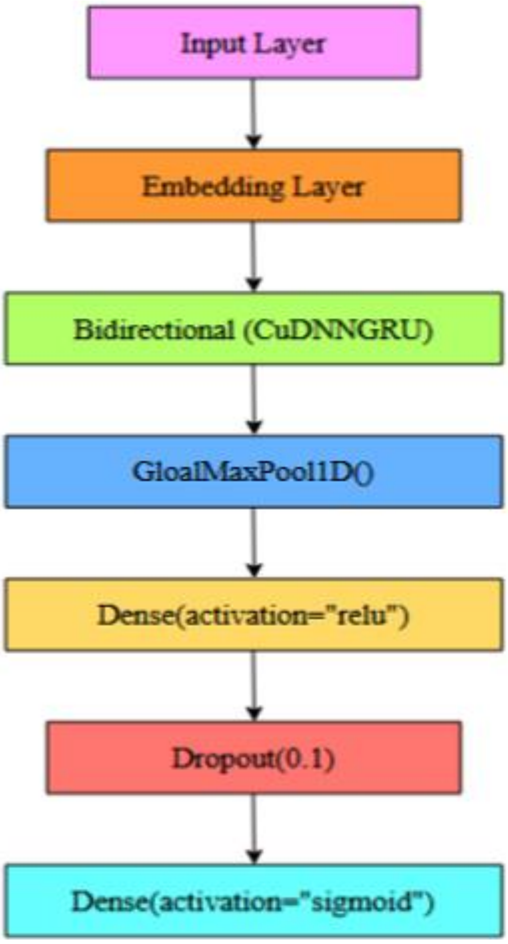

Figure 1.
Architecture of BiGRU implementation.

- Bidirectional CuDNNLSTM: The Bidirectional CuDNNLSTM model is based on Long Short Term Memory (LSTM) units, which are well suited for detecting nuanced, long-range dependencies in text. In the context of insincerity detection, LSTMs are advantageous for handling subtle shifts in tone, sarcasm, or manipulation that may span across the entire question rather than being localized to a single phrase. The architecture of the proposed Bidirectional CuDNNLSTM model is illustrated in Figure 2.

Each LSTM unit maintains a memory cell c_t, updated through input, forget, and output gates that determine how much new information is stored, how much prior memory is retained, and how the current state is exposed to the next layer. The update rules are as represented through Equation 6:

$$\begin{gathered} f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \\ i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ \tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \\ c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \\ o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ h_t = o_t \odot \tanh(c_t) \end{gathered} \quad (6)$$

Here, the LSTM dynamically adjusts its memory to preserve earlier information if it's contextually important, such as a sarcastic cue at the beginning of a question that only becomes meaningful later.

As with the GRU based model, bidirectional processing is employed to capture both left and right contextual cues. The hidden states from forward and backward passes are concatenated in the same way as was done in Bidirectional CuDNNGRU Equation 4.

The output sequence h^{bi}_t is subjected to GlobalMaxPooling1D, allowing the model to focus on the most prominent linguistic features, such as emotionally loaded or polarized phrases. The pooled features are processed through a ReLU activated dense layer, regularized with dropout (rate = 0.1), and finally passed to a sigmoid output layer for binary classification, as defined in Equation 5.

This architecture is particularly effective for capturing complex sentence structures and context shifts, both of which are common in insincere or misleading questions posed in user generated content platforms like Quora.

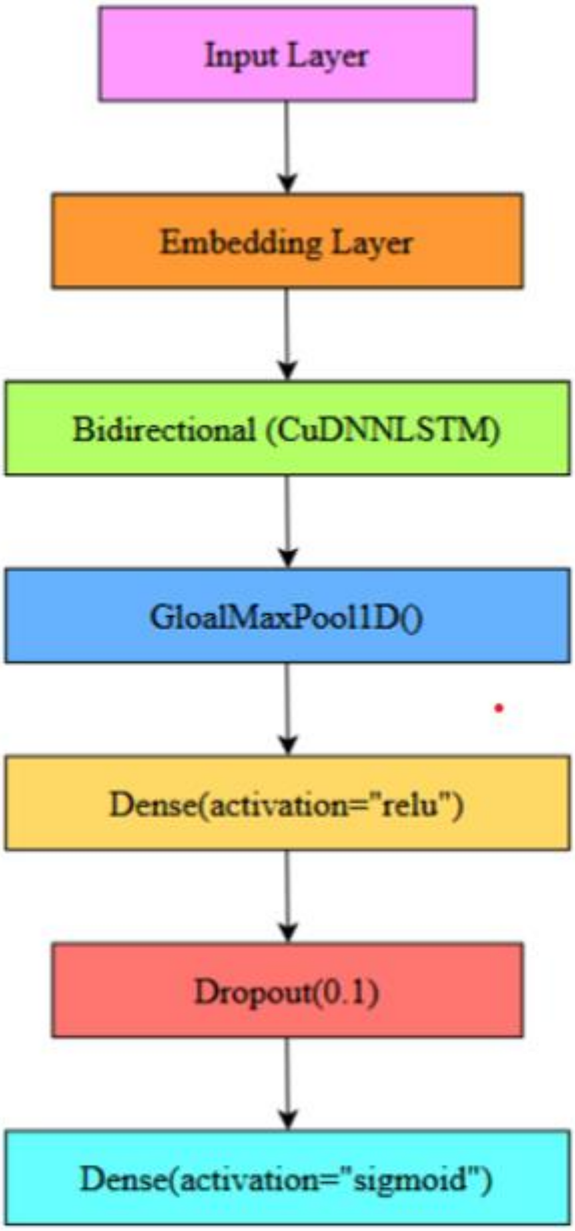

Figure 2.
Architecture of BiLSTM implementation.

- *Capsule Network:* The third model combines traditional sequential layers with Capsule Networks (CapsNet) to better capture both the flow of the text and its underlying structure. Unlike regular neural networks that use pooling and may lose important positional information, Capsule Networks are designed to preserve relationships between features, something that's really useful when you're trying to detect subtle signs of insincerity, like sarcasm or exaggerated phrasing. The architecture of the proposed CapsNet model is illustrated in Figure 3.

We start by embedding each token in the input question $X = \{x_1, x_2, . . . , x_n\}$ into a dense vector space using an embedding layer. These embedded vectors are then passed through two parallel recurrent layers, a Bidirectional LSTM and a Bidirectional GRU. We chose to use both because they capture different aspects of the input: LSTMs are good at holding onto long term dependencies and sentence structure, while GRUs are more efficient and focus on short term transitions. By combining them, the model can capture a broader range of patterns that might signal insincerity.

The outputs from the LSTM and GRU layers are each passed through two types of pooling: GlobalMaxPooling1D, which picks up on the strongest features (like emotionally charged words), and GlobalAveragePooling1D, which gives a general sense of the sentence by averaging across time steps. We concatenate both pooled outputs to create a rich feature vector that balances peak signals with overall context.

This combined feature representation is then passed into a Capsule Layer. Unlike dense layers, capsules process features as vectors rather than scalars, which allows them to encode both the presence and orientation of features. Each capsule transforms its input using a learned weight matrix, and the output for capsule j is calculated as in Equation 7:

$$u_j = \sum_i c_{ij} W_{ij} u_i \tag{7}$$

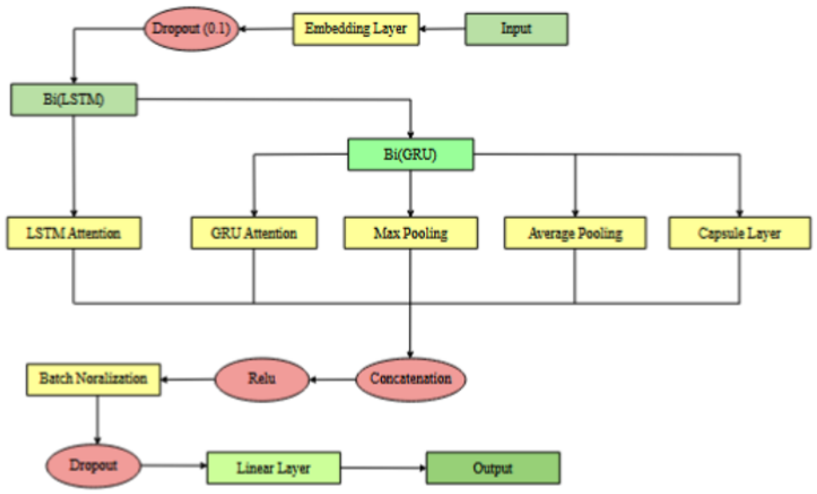

Figure 3.
Architecture of CapsNet Implementation.

Here, u_i is the input from a lower level capsule or feature vector, W_{ij} is the transformation matrix, and c_{ij} are routing coefficients that decide how strongly each capsule influences another. This routing helps the model recognize part whole relationships in phrasing, useful when trying to catch sarcastic or misleading sentence structures.

To ensure stable outputs, we apply a squashing function that compresses the capsule vectors while keeping their direction intact as shown in Equation 8:

$$v_j = \frac{\|u_j\|^2}{1+\|u_j\|^2} \cdot \frac{u_j}{\|u_j\|}$$
(8)

This function scales the output so its length stays between 0 and 1, which can be interpreted as the likelihood that a particular feature is present.

After that, the capsule outputs are flattened and passed through a fully connected dense layer with ReLU activation to learn higher order interactions. We apply a dropout layer with a rate of 0.1 to prevent overfitting. Finally, a sigmoid activated output layer produces a probability $\hat{y} \in [0, 1]$ that reflects how likely the question is insincere shown through Equation 9:

$$\hat{y} = \sigma(W^T v + b)$$
(9)

where v is the flattened capsule output and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function.

Altogether, this model offers a flexible and robust way to capture the kinds of patterns that go beyond simple word meaning like emotional tone, phrasing style, or manipulative sentence structures, making it especially useful for identifying insincerity in online questions.

- *BERT-based Model:* The final architecture explored in this study leverages the BERT-base model, a transformer based language representation model known for its ability to capture deep contextual understanding from text. BERT processes text bidirectionally, allowing it to understand both the left and right context of each word simultaneously. It was originally pretrained on large scale corpora through masked language modeling and next sentence prediction tasks, which equips it with a general understanding of language.

In our setup for insincerity detection, each question is tokenized and formatted during preprocessing, where special tokens such as [CLS] (classification token) and [SEP] (separator token) are added to the input sequence. The tokenized input is then passed through BERT's transformer encoder layers, where self-attention mechanisms compute context aware embeddings for every token in the question.

To classify whether a question is sincere or not, we extract the embedding corresponding to the [CLS] token, denoted $h_{[\mathrm{CLS}]}$—as it serves as a holistic representation of the entire input. This vector is passed through a dense layer with a sigmoid activation to obtain the final prediction depicted through Equation 10:

$$\hat{y} = \sigma(W^T h_{[\mathrm{CLS}]} + b)$$
(10)

Here, W and b are the learnable weight matrix and bias of the classification layer, and $\sigma(x) = \frac{1}{1+e^{-x}}$is the sigmoid activation function that outputs a probability between 0 and 1. The value $\hat{y}$ reflects the model's estimated likelihood that the question is insincere.

To adapt BERT to the specifics of our task, we fine-tune it using binary cross-entropy loss as in Equation 11:

$$\mathrm{L} = -y \log(\hat{y}) - (1-y)\log(1-\hat{y})$$
(11)

where $y \in \{0, 1\}$ is the true label, with 1 indicating insincerity and 0 representing sincerity. During training, BERT learns to recognize subtle linguistic cues such as sarcasm, rhetorical exaggeration, emotionally charged phrasing, or manipulative language—patterns that are often characteristic of insincere questions in online discourse. The architecture of the proposed BERT-Based model i illustrated in Figure 4.

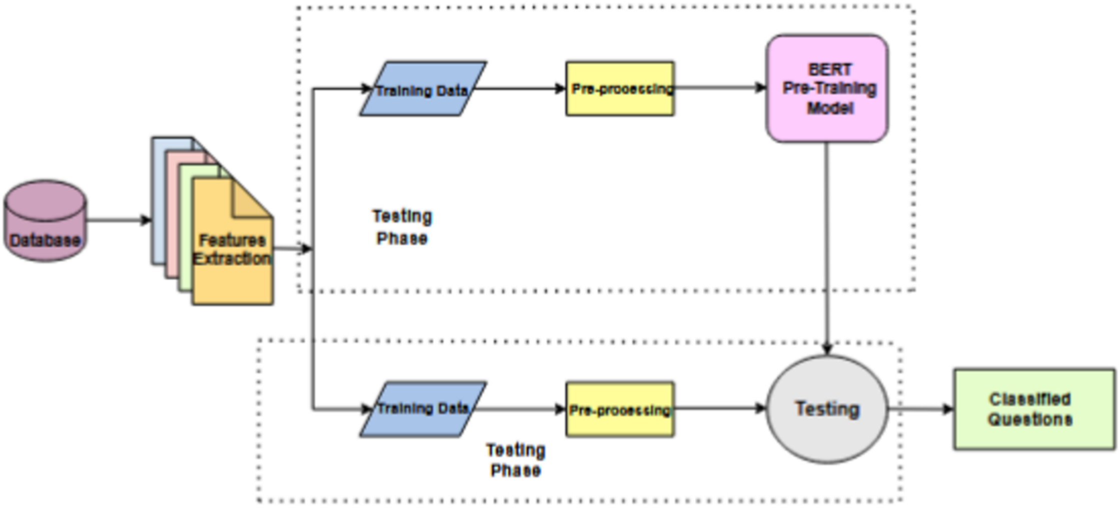

Figure 4.
Architecture of BERT-Based Implementation.

4. Experimental Setup

The Following Sections gives the overview of the experimental setup of the proposed approach.

4.1. Dataset Analysis

The proposed work focuses on detecting insincere questions using Quora's dataset from Kaggle. Quora publicly released its first dataset in 2017. The training dataset contains 1,306,122 samples, while the test dataset consists of 375,806 samples.

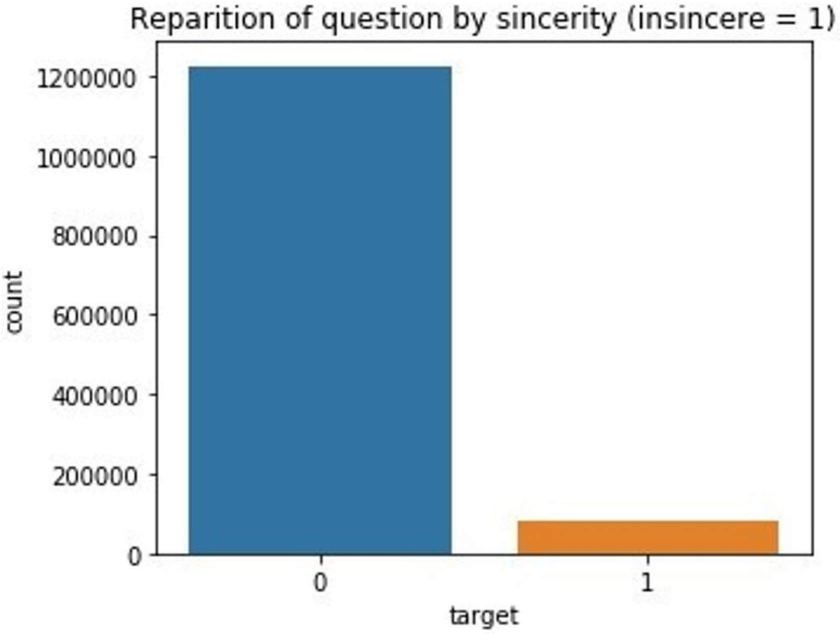

Figure 5.
Target Distribution of Dataset.

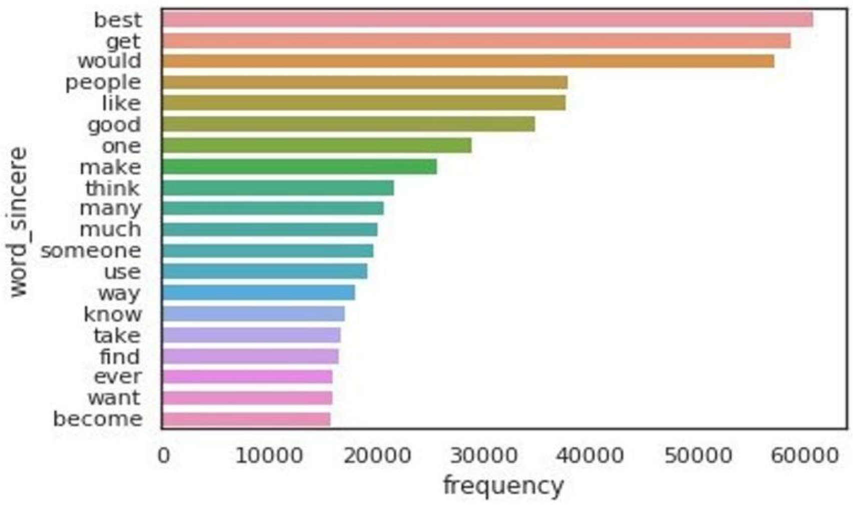

Figure 6.
Word Frequency Distribution for Sincere Questions.

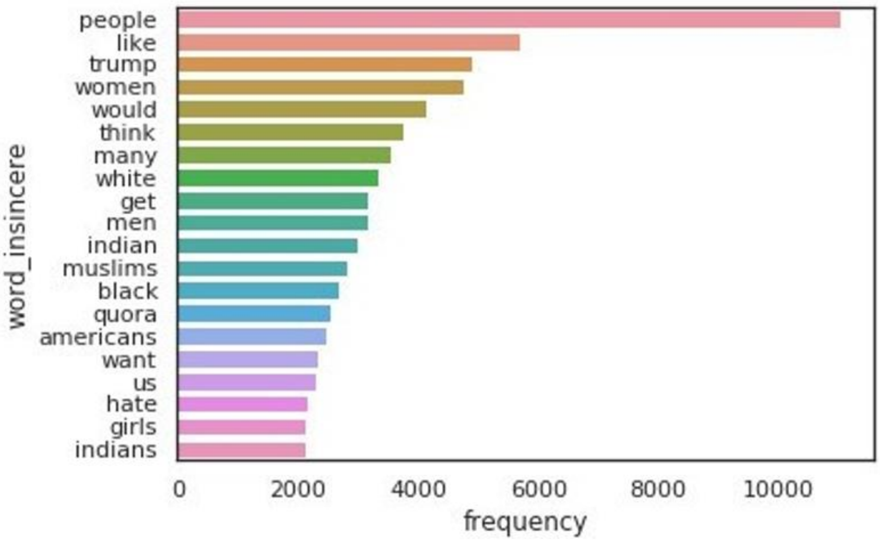

Figure 7.
Word Frequency Distribution for Insincere Questions.

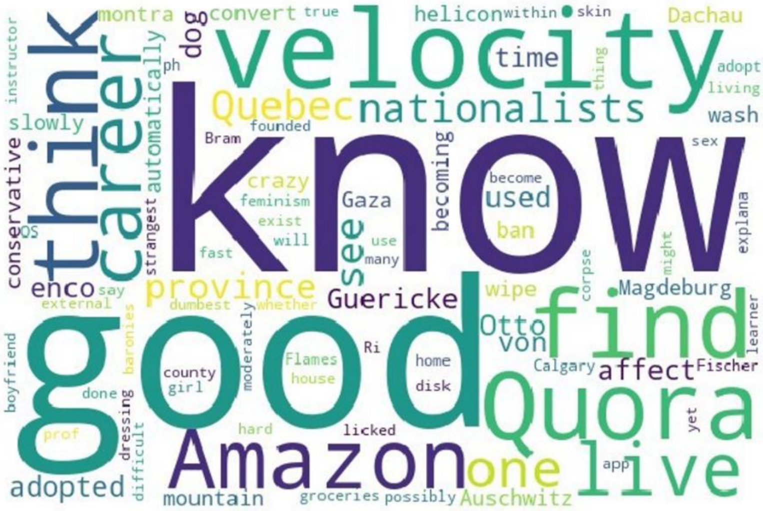

Figure 8.
Word Cloud for Sincere Questions.

The training dataset includes:

- Qid: A unique identifier for each question.
- Question_text: The actual text of the question.
- Target: A binary label, where **0** indicates a sincere question and **1** represents an insincere question.

The dataset analysis provides key insights into the characteristics of both sincere and insincere questions based on their word distributions and thematic content.

Target Distribution: As shown in Figure 5, the dataset predominantly comprises sincere questions (target = 0), with a relatively smaller proportion of insincere questions (target = 1). This distribution highlights the realistic occurrence of insincerity in user generated content. The visualization serves as a foundation for understanding how insincerity is represented within the dataset.

Word Frequency in Sincere Questions: Figure 6 displays the word frequency distribution for sincere questions. Words like *best*, *get*, *would*, *people*, and *like* appear frequently, reflecting the nature of inquiries that are straightforward and often constructive. These words align with the characteristics of sincere questions, which typically focus on seeking advice, solutions, or opinions. The frequency analysis aids in understanding the linguistic patterns prevalent in sincere content.

Word Frequency in Insincere Questions: Figure 7 presents the word frequency distribution for insincere questions. Words such as people, like, trump, women, and think dominate the chart, reflecting the language often used in questions with potentially divisive, politically charged, or biased undertones. This highlights how insincere questions often include terminology that may trigger or exaggerate controversy, reflecting a pattern of negative or provocative engagement. The frequency analysis helps identify key linguistic markers indicative of insincere discourse.

Word Cloud for Sincere Questions: The word cloud in Figure 8 visualizes the vocabulary used in sincere questions. Prominent words such as know, good, find, live, and career suggest that sincere questions often revolve around personal growth, knowledge, and genuine inquiry. This representation underscores the contrast between sincere and insincere content, with sincere questions generally focusing on practical, constructive, and meaningful topics.

Word Cloud for Insincere Questions: The word cloud in Figure 9 highlights the vocabulary used in insincere questions. Prominent words such as American, people, choice, Muslim, and liberal suggest that insincere questions often focus on polarizing or controversial topics. This thematic bias is a critical aspect of insincerity, where the intent is often to provoke, misinform, or incite. The visual representation underscores the semantic differences between sincere and insincere questions

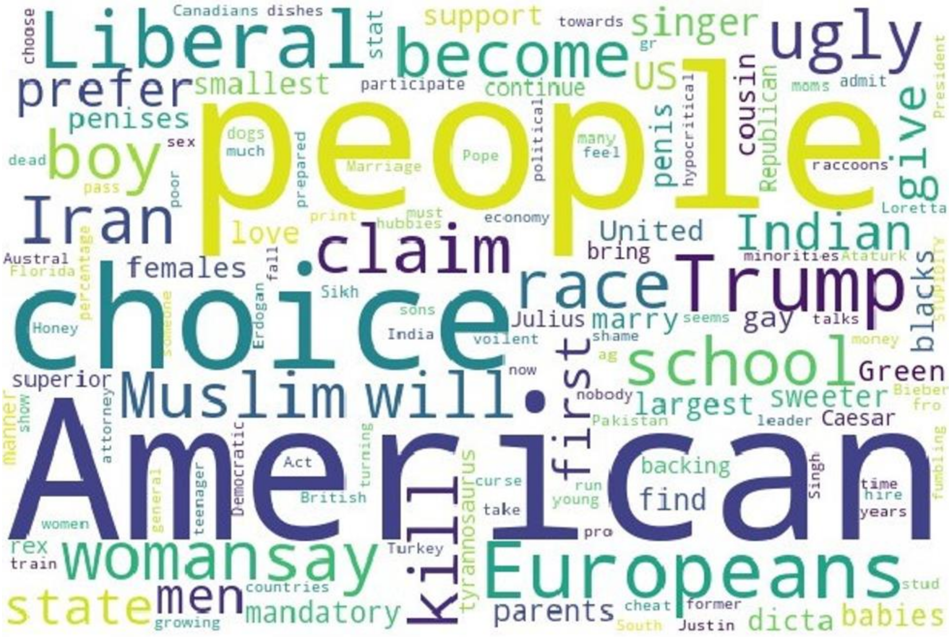

Figure 9.
Word Cloud for Insincere Questions.

4.2. Implementation Details

The proposed model is implemented using Keras with TensorFlow as the backend, leveraging TensorFlow's computational efficiency and Keras's streamlined model development capabilities. The training process is conducted over 15 epochs, ensuring the model undergoes multiple iterations to refine its learning. The dataset is split into 98% for training and 2% for validation, allowing the model to learn from the majority of the data while still being evaluated on a small validation set to monitor generalization. At the end of each epoch, the model's performance is assessed on the validation set, and the best performing checkpoint is retained. This strategy ensures that the most optimal model, achieving the best validation performance, is selected for final inference on the test dataset.

Table 3.
Hyperparameter Settings for the Experiment.

Parameter	Value
Maximum length of sequence	74
Batch size	60
Epochs	15
Optimizer	BERTAdam
Learning rate	5e-6
Dropout rate	0.1
Loss function	Sigmoid Binary Cross Entropy Loss

4.3. Hyperparameters Setting

The hyperparameters for this experiment were meticulously selected through an extensive search process, ensuring optimal model performance. The selection process involved evaluating various configurations, including sequence length, batch size, optimizer, learning rate, and regularization techniques, to determine the most effective combination for detecting insincere text. Given the sensitivity of deep learning models to hyperparameter choices, multiple experiments were conducted to assess the impact of different settings, ultimately leading to the final configuration. The BERTAdam optimizer was chosen due to its effectiveness in fin tuning transformer based while maintaining computational efficiency. The batch size was fixed at 60, striking a balance between stability and memory constraints. To prevent overfitting, a dropout rate of 0.1 was applied, enhancing the model's robustness. The loss function, Sigmoid Binary Cross Entropy Loss, was employed to handle the binary classification task effectively. These hyperparameter choices align with best practices in transformer based text classification, ensuring a well regularized and high performing model. Table 3 provides a detailed overview of the selected hyperparameters.

4.4. Evaluation Metrics

To assess the performance of our model in detecting insincere questions, we employed standard classification evaluation metrics, including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive analysis of the model's effectiveness in distinguishing between sincere and insincere questions.

Accuracy: This metric measures the overall correctness of the model by computing the ratio of correctly classified instances to the total number of instances. It is defined as Equation 12:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

where *TP* (True Positives) represents correctly classified insincere questions, *TN* (True Negatives) denotes correctly classified sincere questions, *FP* (False Positives) refers to sincere questions misclassified as insincere, and *FN* (False Negatives) indicates insincere questions misclassified as sincere.

Precision: Precision evaluates the model's ability to correctly identify insincere questions among all instances predicted as insincere. It is defined as Equation 13:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{13}$$

Recall: Also known as sensitivity, recall measures the model's effectiveness in identifying insincere questions from the total actual insincere questions. It is given by Equation :

$$\text{Recall} = \frac{TP}{TP + FN} \tag{14}$$

F1-score: The F1-score is the harmonic mean of precision and recall, balancing both metrics to provide a single performance measure. It is defined as Equation 15:

These metrics collectively ensure a robust evaluation of the model, capturing both correctness and misclassification tendencies. While accuracy provides a general performance measure, precision and recall highlight the model's effectiveness in detecting insincere questions, and F1-score balances the trade off between them.

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

5. Results

In this section, we present the outcomes of our experiments, demonstrating the effectiveness of the proposed model in detecting insincere questions. The evaluation metrics discussed earlier are employed to provide a comprehensive analysis of the model's performance across various scenarios.

To evaluate the effectiveness of different architectures in detecting insincere questions, multiple models were tested, including Bidirectional CuDNNGRU, Bidirectional CuDNNLSTM, Capsule Networks, and BERT. The BiGRU model demonstrated a solid performance with an accuracy of 94.68%, indicating its ability to classify questions correctly in most cases. Its precision of 0.6936 reflects a reasonable success in identifying true insincere predictions while minimizing false positives. However, the recall of 0.5663 reveals that the model struggles to capture a significant proportion of actual insincere questions, leading to an F1-score of 0.6533, which provides a moderate balance between precision and recall.

The BiLSTM model showed slightly lower accuracy at 94.0%, signifying a marginally reduced overall correctness compared to BiGRU. Its precision of 0.6897 is comparable to that of BiGRU, but the recall of 0.5504 indicates it captures fewer insincere questions, missing nearly 45% of them. Consequently, the F1-score of 0.6443 highlights the BiLSTM's slightly weaker balance between precision and recall, making it less effective for this task compared to BiGRU.

The Capsule Network emerged as a more precise model, achieving an accuracy of 95.70% and the highest precision of 0.7248 among the recurrent architectures tested. This indicates its strength in correctly classifying insincere questions while minimizing false positives. However, the model's recall of 0.4924 suggests a significant limitation in identifying all actual insincere questions, capturing less than half of them. As a result, its F1-score of 0.6466, while similar to BiGRU and BiLSTM, does not represent a significant improvement in overall performance due to the tradeoff between precision and recall.

The BERT model consistently outperformed the other architectures across all metrics. It achieved the highest accuracy of 96.39%, demonstrating superior correctness in classifying questions. With a precision of 0.7089, BERT effectively minimizes false positives while maintaining strong insincere predictions. More notably, its recall of 0.7265 underscores its ability to identify a larger proportion of actual insincere questions compared to other models. The resulting F1-score of 0.721 reflects the best balance between precision and recall, making BERT the most effective and reliable model for insincerity detection.

Overall, the analysis highlights BERT as the most robust architecture for this task, delivering superior results in terms of accuracy, precision, recall, and F1-score. While BiGRU and BiLSTM offer moderate performance, their lower recall and F1-scores make them less suitable for accurately identifying insincere questions. The Capsule Network, despite its precision, struggles with recall, limiting its applicability. These findings underscore the advantage of transformer based models like BERT in addressing complex NLP tasks such as insincerity detection. Table 4 presents the performance results of all tested models.

Table 4.
Performance Comparison of Tested Models.

Model	Accuracy	Precision	Recall	F1-score
BiCuDNNGRU	0.9468	0.6936	0.5663	0.6533
BiCuDNNLSTM	0.9400	0.6897	0.5504	0.6443
Capsule Network	0.9570	0.7248	0.4924	0.6466
BERT	0.9639	0.7089	0.7265	0.7210

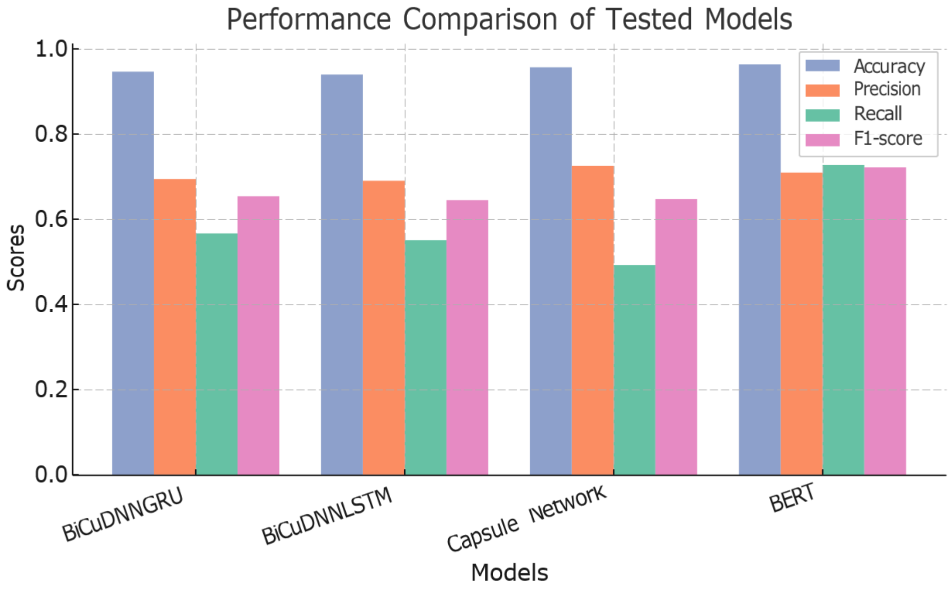

Figure 10.
Performance Comparison of tested models.

Figure 10 illustrates the performance comparison of all the tested models in terms of accuracy, precision, recall, and F1- score.

In recent research on insincerity detection in user generated content, several models have been explored to enhance performance. Mishra and Kumar [50] employed a Convolutional Neural Network (CNN) combined with GloVe embeddings, which significantly improved vocabulary coverage. This approach achieved an impressive accuracy of 0.902, showcasing the importance of embedding techniques in capturing subtle nuances of language that could indicate insincerity.

In contrast, Chakraborty, et al. [51] utilized DeBERTa, a variant of the BERT model, to detect insincerity. Their method outperformed others, achieving an accuracy of 0.95. The enhanced attention mechanism in DeBERTa allowed it to capture deeper contextual relationships in the text, making it effective in understanding the subtleties of insincere language. Similarly, Rachha and Vanmane [44] explored RoBERTa and ALBERT for insincerity detection, with RoBERTa yielding the highest accuracy of 0.955, while ALBERT followed with an accuracy of 0.949.

In comparison to these models, our approach employs BERT, which has consistently set new benchmarks for insincerity detection. Our results indicate that BERT achieved an accuracy of 0.96, surpassing all previously mentioned models, including DeBERTa and RoBERTa. While CNN-GloVe model excelled in vocabulary coverage, BERT's context aware attention mechanism enables it to better understand complex linguistic cues indicative of insincerity, such as manipulative or inflammatory phrasing. Our results also demonstrate that BERT's pretrained language representations, finetuned on the specific task of insincerity detection, provide superior accuracy compared to models like RoBERTa and ALBERT. This highlights BERT's robustness in capturing intricate contextual information and its ability to generalize effectively across diverse types of user generated content, leading to improved detection of insincere behavior. Table V provides a comparison of our approach with other state-of-the-art models for insincerity detection, showcasing the superior accuracy of the proposed approach. Figure 11 illustrates the performance comparison of various models for insincerity detection, highlighting the superior accuracy of the proposed approach.

Table 5.
Performance Comparison of Different Models for Insincerity Detection.

Model	Accuracy
CNN + GloVe [50]	0.902
DeBERTa [51]	0.95
RoBERTa [44]	0.955
ALBERT [44]	0.949
Proposed Approach	0.96

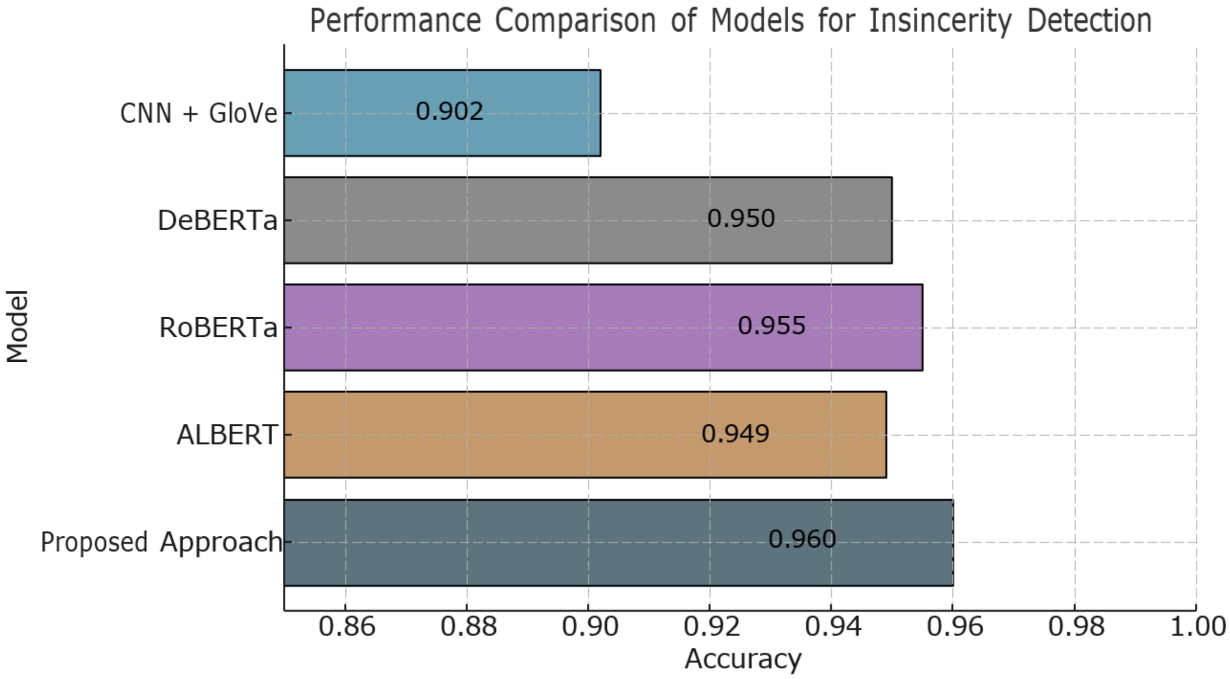

Figure 11.
Performance Comparison Different Models.

6. Conclusion

This study demonstrates the effectiveness of transformer based models, particularly BERT, in detecting insincere questions within textual data. Compared to traditional recurrent models such as BiCuDNNGRU and BiCuDNNLSTM, and even advanced architectures like Capsule Networks, BERT exhibited superior performance across all key evaluation metrics. The results reinforce the growing consensus in the NLP community regarding the effectiveness of pretrained transformer based models for nuanced language understanding tasks.

Prior studies have explored various deep learning and transformer based models for insincerity detection. The performance comparison highlights that models such as CNN with GloVe and ALBERT achieved competitive results, but the proposed approach outperformed them, achieving the highest accuracy of 0.96. Compared to RoBERTa and DeBERTa, which achieved accuracy scores of 0.955 and 0.95 respectively, the proposed method exhibited marginal but meaningful improvements, indicating that optimized hyperparameter selection and fine-tuning strategies played a crucial role in enhancing performance. These results align with previous findings that fine-tuning transformer based models with carefully chosen hyperparameters and preprocessing strategies significantly improves classification accuracy.

A closer examination of precision, recall, and F1-score provides insights into model behavior. While Capsule Networks demonstrated promising accuracy, its recall was significantly lower than that of BERT, indicating a tendency to misclassify some insincere questions. In contrast, BERT achieved a balanced performance across all metrics, effectively identifying both sincere and insincere questions without overfitting to a specific class. The higher recall value of 0.7265 suggests that the model captures a broader range of insincerity patterns, making it a more reliable solution for real world applications.

The findings of this research have several implications. The ability to automatically detect insincere questions can be integrated into online platforms such as Quora, Stack Overflow, or community driven forums to enhance content moderation and prevent the spread of misinformation. Additionally, the methodology used in this study, leveraging pretrained transformers for binary classification tasks, can be extended to other text classification problems, such as hate speech detection and fake news identification.

This study presented a transformer based approach for insincerity detection in textual data using the Quora dataset. The experimental analysis revealed that BERT outperforms traditional deep learning models, including BiCuDNNGRU, BiCuDNNLSTM, and Capsule Networks, across all key evaluation metrics. The proposed model achieved the highest accuracy of 0.96, surpassing other state-of-the-art methods such as RoBERTa, ALBERT, and DeBERTa. By leveraging pretrained contextual embeddings and fine-tuning strategies, the model effectively captures the subtle linguistic features associated with insincere text, reinforcing the viability of transformer based models for automated content moderation and other text classification tasks.

References

[1] K. Shu, D. Mahudeswaran, S. Wang, and H. Liu, "Hierarchical propagation networks for fake news detection: Investigation and exploitation," in *Proceedings of the International AAAI Conference on Web and Social Media*, 2020, vol. 14, pp. 626-637.

[2] V. Umarani, A. Julian, and J. Deepa, "Sentiment analysis using various machine learning and deep learning techniques," *Journal of the Nigerian Society of Physical Sciences,* vol. 3, no. 4, pp. 385-394, 2021. https://doi.org/10.46481/jnsps.2021.308

[3] M. Lupei and M. Shliakhta, "The fake news detection model explanation and infrastructure aspects," presented at the International Conference on Optimization and Data Science in Industrial Engineering. Springer, 2023.

[4] E. Aïmeur, S. Amri, and G. Brassard, "Fake news, disinformation and misinformation in social media: A review," *Social Network Analysis and Mining,* vol. 13, no. 1, p. 30, 2023. https://doi.org/10.1007/s13278-023-01028-5

[5] L. Yuan, H. Jiang, H. Shen, L. Shi, and N. Cheng, "Sustainable development of information dissemination: A review of current fake news detection research and practice," *Systems,* vol. 11, no. 9, p. 458, 2023. https://doi.org/10.3390/systems11090458

[6] X. Zhou and R. Zafarani, "A survey of fake news: Fundamental theories, detection methods, and opportunities," *ACM Computing Surveys,* vol. 53, no. 5, pp. 1-40, 2020. https://doi.org/10.1145/3395046

[7] V. L. Rubin, N. Conroy, Y. Chen, and S. Cornwell, "Fake news or truth? using satirical cues to detect potentially misleading news," in *Proceedings of the Second Workshop on Computational Approaches to Deception Detection*, 2016, pp. 7-17.

[8] A. Reyes and P. Rosso, "Mining subjective knowledge from customer reviews: A specific case of irony detection," in *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011)*, 2011, pp. 118-124.

[9] Y. Chen, Y. Zhou, S. Zhu, and H. Xu, "Detecting offensive language in social media to protect adolescent online safety," presented at the 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing, IEEE, 2012.

[10] C. Van Hee, G. Jacobs, C. Emmery, B. Desmet, E. Lefever, and B. Verhoeven, "Automatic detection of cyberbullying in social media text," *PloS One,* vol. 13, no. 10, p. e0203794, 2018. https://doi.org/10.1371/journal.pone.0203794

[11] R. Y. Lau, S. Y. Liao, R. C. W. Kwok, K. Xu, Y. Xia, and Y. Li, "Text mining and probabilistic language modeling for online review spam detection," *ACM Transactions on Management Information Systems,* vol. 2, no. 4, pp. 1-30, 2012. https://doi.org/10.1145/2070710.207071

[12] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in *Proceedings of the 25th International Conference on World Wide Web*, 2016, vol. 2016, pp. 145–153.

[13] M. M. Ben Ismail and O. Bchir, *Insult detection in social network comments using possibilistic based fusion approach. In Computer and Information Science*. Cham: Springer International Publishing, 2014.

[14] H. J. Escalante, E. Villatoro-Tello, A. Juárez, M. Montes, and L. Villaseñor-Pineda, "Sexual predator detection in chats with chained classifiers," in *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2013, pp. 46-54.

[15] S. T. Indra, L. Wikarsa, and R. Turang, "Using logistic regression method to classify tweets into the selected topics," presented at the 2016 International Conference on Advanced Computer Science and Information Systems, IEEE, 2016.

[16] E. F. Unsvåg and B. Gambäck, "The effects of user features on Twitter hate speech detection," in *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, 2018, pp. 75-85.

[17] A. Al-Khatib and S. R. El-Beltagy, "Emotional tone detection in arabic tweets," presented at the International Conference on Computational Linguistics and Intelligent Text Processing, Cham: Springer International Publishing, 2017.

[18] K. Ravi and V. Ravi, "A novel automatic satire and irony detection using ensembled feature selection and data mining," *Knowledge-Based Systems,* vol. 120, pp. 15-33, 2017. https://doi.org/10.1016/j.knosys.2016.12.018

[19] R. Chittari, M. S. Nistor, D. Bein, S. Pickl, and A. Verma, "Classifying sincerity using machine learning," presented at the ITNG 2022 19th International Conference on Information Technology-New Generations, Cham: Springer International Publishing, 2022.

[20] R. Kumar, A. Kumar, M. Gupta, and B. Chauhan, "Quora based insincere content classification & detection for social media using machine learning," presented at the 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), IEEE, 2021.

[21] T. Nguyen and P. Meesad, "A study of predicting the sincerity of a question asked using machine learning," in *Proceedings of the 2021 5th International Conference on Natural Language Processing and Information Retrieval*, 2021, pp. 129-134.

[22] D. Omarova, F. A. Dael, I. Shayea, G. Abitova, and E. Sailaukhanov, "Detecting questions in online communities: A machine learning approach," presented at the 2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (CICN), IEEE., 2024.

[23] D. Goyal, S. K. Mishra, and V. K. Rai, "A multimodal framework for satire vs. sarcasm detection," presented at the 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE., 2024.

[24] T. Lappas, "Fake reviews: The malicious perspective," presented at the International Conference on Application of Natural Language to Information Systems, Springer, 2012.

[25] J. Malbon, "Taking fake online consumer reviews seriously," *Journal of Consumer Policy,* vol. 36, no. 2, pp. 139-157, 2013. https://doi.org/10.1007/s10603-012-9216-7

[26] H. Li, Z. Chen, B. Liu, X. Wei, and J. Shao, "Spotting fake reviews via collective positive-unlabeled learning," presented at the 2014 IEEE International Conference on Data Mining, IEEE, 2014.

[27] E. Papegnies, V. Labatut, R. Dufour, and G. Linares, "Graph-based features for automatic online abuse detection," presented at the International Conference on Statistical Language and Speech Processing, Cham: Springer International Publishing, 2017.

[28] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? predictive features for hate speech detection on twitter," in *Proceedings of the NAACL Student Research Workshop*, 2016, pp. 88–93.

[29] A. Mungekar, N. Parab, P. Nima, and S. Pereira, *Quora insincere question classification*. Ireland: National College of Ireland, 2019.

[30] B. Gaire, B. G. Rijal, D., S. Sharma, and N. Lamichhane, "Insincere question classification using deep learning," *International Journal of Scientific & Engineering Research,* vol. 10, pp. 2001–2004, 2019.

[31] D. Mediratta and N. Oswal, "Detect toxic content to improve online conversations," *arXiv preprint arXiv:1911.01217,* 2019. https://doi.org/10.48550/arXiv.1911.01217

[32] A. Ranganathan, H. Ananthakrishnan, D. Thenmozhi, and C. Aravindan, "Classification of insincere questions using SGD optimization and SVM classifiers," in *Proceedings of the Forum for Information Retrieval Evaluation (FIRE 2019) Working Notes, CEUR-WS*, 2019, pp. 463–467.

[33] V. Mujadia, P. Mishra, and D. M. Sharma, "Classification of insincere questions with ML and neural approaches," in *Proceedings of the Forum for Information Retrieval Evaluation (FIRE 2019) Working Notes, CEUR-WS*, 2019, pp. 451–455.

[34] M. A. Al-Ramahi and I. Alsmadi, "Using data analytics to filter insincere posts from online social networks: A Case Study: Quora insincere questions," presented at the Proceedings of the 53rd Hawaii International Conference on System Sciences (HICSS-53), 2020.

[35] R. Kshirsagar, T. Cukuvac, K. McKeown, and S. McGregor, "Predictive embeddings for hate speech detection on twitter," *arXiv preprint arXiv:1809.10644,* 2018. https://doi.org/10.48550/arXiv.1809.10644

[36] S. Agrawal and A. Awekar, "Deep learning for detecting cyberbullying across multiple social media platforms," presented at the European Conference on Information Retrieval, Cham: Springer International Publishing, 2018.

[37] N. Lu, G. Wu, Z. Zhang, Y. Zheng, Y. Ren, and K. K. R. Choo, "Cyberbullying detection in social media text based on character-level convolutional neural network with shortcuts," *Concurrency and Computation: Practice and Experience,* vol. 32, no. 23, p. e5627, 2020. https://doi.org/10.1002/cpe.5627

[38] S. J. Bu and S. B. Cho, "A hybrid deep learning system of CNN and LRCN to detect cyberbullying from SNS comments," presented at the International Conference on Hybrid Artificial Intelligence Systems, Cham: Springer International Publishing, 2018.

[39] A. Kumar and G. Garg, "Sarcasm detection using feature-variant learning models," in *Proceedings of ICETIT 2019: Emerging Trends in Information Technology, Cham: Springer International Publishing*, 2019, pp. 683-693.

[40] Y. G. Cheong, A. K. Jensen, E. R. Guðnadóttir, B. C. Bae, and J. Togelius, "Detecting predatory behavior in game chats," *IEEE Transactions on Computational Intelligence and AI in Games,* vol. 7, no. 3, pp. 220-232, 2015.

[41] F. Monti, F. Frasca, D. Eynard, D. Mannion, and M. M. Bronstein, "Fake news detection on social media using geometric deep learning," *arXiv preprint arXiv:1902.06673,* 2019. https://doi.org/10.48550/arXiv.1902.06673

[42] H. Yenala, A. Jhanwar, M. K. Chinnakotla, and J. Goyal, "Deep learning for detecting inappropriate content in text," *International Journal of Data Science and Analytics,* vol. 6, no. 4, pp. 273-286, 2018. https://doi.org/10.1007/s41060-017-0088-4

[43] S. Gottipati, A. Tan, D. Chow, J. Shan, J. Lim, and W. Kiat, "Leveraging profanity for insincere content detection-a neural network approach," presented at the 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), IEEE, 2020.

[44] A. Rachha and G. Vanmane, "Detecting insincere questions from text: A transfer learning approach," *arXiv preprint arXiv:2012.07587,* 2020. https://doi.org/10.48550/arXiv.2012.07587

[45] I. Aslam, M. A. Zia, I. Mumtaz, Q. Nawaz, and M. Hashim, "Classification of insincere questions using deep learning: quora dataset case study," presented at the International Conference on Management Science and Engineering Management, Cham: Springer International Publishing., 2021.

[46] D. Arora, G. Aggarwal, and F. Asif, "Quora question sincerity detection using bert-based framework," presented at the 2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence), IEEE, 2024.

[47] Z. Zhou, H. Guan, M. M. Bhat, and J. Hsu, "Fake news detection via NLP is vulnerable to adversarial attacks," in *Proceedings of the 11th International Conference on Agents and Artificial Intelligence. https://doi.org/10.5220/0007566307940800*, 2019, pp. 794-800.

[48] V. L. Rubin, N. J. Conroy, and Y. Chen, "Towards news verification: Deception detection methods for news discourse," presented at the Hawaii International Conference on System Sciences, 2015.

[49] M. Del Pilar Salas-Zárate, M. A. Paredes-Valverde, M. Á. Rodriguez-García, R. Valencia-García, and G. Alor-Hernández, "Automatic detection of satire in Twitter: A psycholinguistic-based approach," *Knowledge-Based Systems,* vol. 128, pp. 20-33, 2017. https://doi.org/10.1016/j.knosys.2017.04.009

[50] S. Mishra and N. Kumar, "Insincere questions classification using cnn with increased vocabulary coverage of glove embedding," *Journal of The Institution of Engineers (India): Series B,* vol. 104, no. 2, pp. 387-394, 2023. https://doi.org/10.1007/s40031-023-00858-3

[51] S. Chakraborty *et al.*, "Quora insincere questions classification using attention based model," presented at the International Conference on Data Science and Emerging Technologies, Singapore: Springer Nature Singapore, 2022.