Published in the Proceedings of the IEEE TPS 2024:
The Sixth IEEE International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications

Bias Detection and Mitigation in Zero-Shot Spam Classification using LLMs

Hossein Salemi
Humanitarian Informatics Lab
George Mason University
Fairfax, VA, USA
hsalemi@gmu.edu

Anuridhi Gupta
Humanitarian Informatics Lab
George Mason University
Fairfax, VA, USA
agupta29@gmu.edu

Hemant Purohit
Humanitarian Informatics Lab
George Mason University
Fairfax, VA, USA
hpurohit@gmu.edu

***Abstract*—There is a growing number of scams through various communication mediums, including social media, phone calls and messages, search engine advertising, etc. Often these scams are realized via sending spam texts on any of these communication mediums, and therefore, prior research has investigated the task of spam classification to design information filtering systems. However, existing works have explored supervised machine learning techniques primarily, which suffer from the bottleneck of requiring large labeled datasets. Further, the studies are based on platform-specific data and lack critical analyses of biases in the predictive modeling behavior. In this paper, we propose a zero-shot spam classification task, which does not require any labeled data for model training in an unseen domain. We propose a novel method to leverage state-of-the-art large language models (LLMs) in natural language processing (NLP) for this task. We employ this method to analyze zero-shot performance on spam datasets across two communication platforms (YouTube and phone messages) while mitigating biases in the model behavior. Our experimental results demonstrate the strong performance of a LLM-based zero-shot classifier with a goal-oriented prompting strategy. The resulting classifier is platform-agnostic, shows less bias towards data with certain behavioral attributes (e.g., spam content with sadness emotion), and is efficient in minimizing both false positive and false negative errors. The application of this research can inform effective spam filtering systems to prevent scams prevalent in different communication media ultimately.**

***Index Terms*—Online Scams, Spam, Phishing, Large Language Models, Inclusive Cybersecurity**

I. Introduction

The growth of social engineering attacks to deceive or scam others through spam messages such as click baits and phishing has a devastating impact on social and economic aspects of our daily lives. The Consumer Sentinel Network Data Book of 2023 of the U.S. Federal Trade Commission [1] indicates nearly 2.5 million reports of fraud and $10 billion in losses last year. Such attacks include false claims for distressed relatives, romance, investment, etc., and often initiate communication via online (social media) and offline (text messages) platforms.

The act of scamming is realized through spam messaging on different communication platforms by malicious actors such as bots, hackers, and cybercriminals representing a major risk in cybersecurity. Formally, the concept of spam refers to a broad category of unsolicited messages, ranging from fake advertising, malicious URLs, clickbait, objectionable and persuasive content [2]. Researchers have studied spam across diverse communication platforms such as emails [3], phone calls and short messages (SMS) [4], social networking services [5], review forums [6], etc.

Prior research has investigated machine learning-based NLP techniques for detecting spam messages [7], [8]. Researchers explored both manual and automated feature engineering approaches for training a classification model. The deceiving use of behavioral attributes in the language employed by scammers to influence the judgment of message receivers [9] can challenge the performance of classification models, such as by showing persuasion, sadness, urgency, and fear.

The heavy reliance upon supervised machine learning in existing research on spam classification models presents limitations. First, it demands a large labeled dataset to train a model efficiently. Second, it requires a labeled dataset for each platform. Third, the predictive decisions can contain unintended model biases [10] due to data collection, labeling, and feature engineering methods employed in the existing literature for zero-shot spam classification tasks. While there has been considerable exploration to address bias and fairness concerns in machine learning [11], [12], there is not much attention given to the spam classification models, especially concerning behavioral attributes of language in spam [9]. To address these limitations, we make the following contributions:

1) We propose a zero-shot spam classification method using state-of-the-art NLP technology of LLMs [13], which enables a platform-agnostic approach by leveraging the LLMs' knowledge from pre-training on enormous data.
2) We conduct a critical analysis of the unintended bias of models in a spam classification task, based on the behavioral attributes of the language of spam content.
3) Through extensive experimentation on datasets from two platforms (YouTube, SMS), we present novel insights on a goal-oriented prompting strategy for an LLM-based spam classification model to minimize unintended bias and reduce prediction errors.

The rest of the paper is structured as follows. Section II presents the related work, followed by Section III, our methodology for bias detection and mitigation. Section IV describes the experimental setup, followed by results in Section V, before concluding with limitations and future work in Section VI.

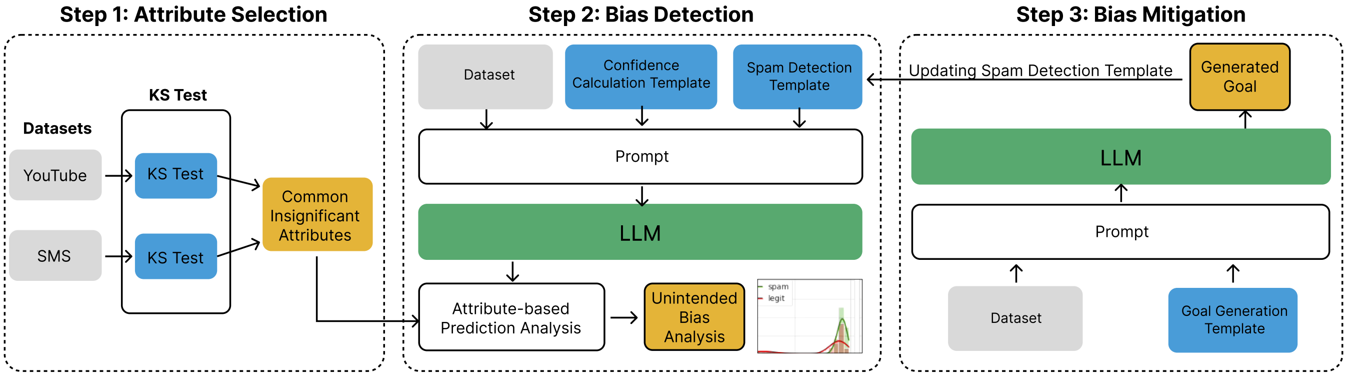


Fig. 1: Summary of the proposed approach for bias detection and mitigation in spam classification.

II. Related Work

We summarize the literature in two relevant directions.

A. *Spam classification tasks and methods*

An increased reliance on technology in the post-pandemic years has enabled a breeding ground for spamming, phishing, and ultimately, cyber crimes. To tackle this problem, there has been extensive exploration to leverage machine learning models that can automatically detect spam across different platforms and warn the users. A vast majority of existing work on spam detection focuses on supervised machine learning [2], [7], [14]. Current work has explored different types of spam classification problems such as binary spam classification [15] and multi-class spam classification [16] across different platforms, although with limited explorations on communication mediums of social networks and SMS in contrast to emails and web URLs [14]. For instance, recent surveys [7] provide examples of machine-learning models for detecting SMS-based scams. Further, prior research has also explored the anomaly or outlier detection [17] method, which is a popular technique to find patterns in data that exhibit unexpected behavior [14]. For designing an efficient spam classifier or filtering system, feature extraction is a crucial part of the process. For example, [18] proposes two spam review detection models: first, which utilizes thirteen different spammers' behavioral features, and second, which uses the linguistic features of the content in identifying spam reviews. Both these approaches outperformed existing models regarding the accurate identification of spam reviews. However, crafting manual features is a complex task. Thus, recent works have started exploring the use of deep learning and LLM-based approaches. [19] discusses malicious prompting in LLMs to generate phishing scams and then further proposes a BERT-based automated spam detection model that is effective in detecting these malicious prompts at an early stage to prevent the generation of spam content. [20] highlights the utility of LLMs in providing efficiency in identifying common phishing and scam emails. Our proposed work complements these recent explorations by investigating a spam detection task in a zero-shot setting on cross-platform datasets, which is lacking in the current literature.

B. *Bias and Fairness of ML models*

It is not sufficient for a machine learning model to only detect spam messages. It is essential to question the model's internal mechanism to reason behind its predictions, and why a model is classifying a message as spam or legit. This crucial understanding can uncover the intrinsic bias of the model. Extensive recent works in machine learning establish the existence of unintended bias in models. A recent survey [21] provides an in-depth analysis of different sources of model biases that exist in applications and how fairness in machine learning can help eliminate the existing bias. [22] evaluates existing bias mitigation techniques and provides insights on diverse fairness metrics that have been employed in the past. [23] provides an overview of different types of biases, where they possibly come from and how to best mitigate them. Further, bias can make certain groups more susceptible to phishing and scam attacks especially when the training data has under-inclusion of language factors that are likely to deceive vulnerable populations such as the elderly and people with autism spectrum disorder (ASD) [24], [25]. For instance, [24] shows that users with ASD post a greater number of social media posts with "fear", "anxiety", and "paranoia", which could be some of the attributes of the language employed in the content of spam messages [9]. Similarly, people who have physical and/or mental health conditions are more prone to engage with fraudulent content due to their situational vulnerabilities such as loneliness, social isolation, and anxiety [26]. Recent works [11], [27] delve into the significance of fairness and bias metrics, as the emerging threat of generative AI-based automated spam and fraud are at the doorstep. While this work is influential in our current approach, the existing literature lacks a comprehensive cross-platform study (YouTube, SMS) for bias analysis. Our study complements these existing works by addressing the underlying unintended bias for models developed for the task of spam classification.

III. Methodology

Here, we describe our methodology for detecting and mitigating the biases of LLMs [13] in a zero-shot spam classification task for distinguishing spam versus legit messages, as summarized in Fig. 1. In the *first* step, we analyze a given dataset to uncover the various behavioral attributes present in the language used in messages, which are not significantly discriminating features to classify spam messages. In the *second* step, these attributes are studied as the potential sources of unintended bias in LLMs. In particular, for the subsets of data extracted based on the values of each attribute, we explore the performance of the LLM model on the spam classification task. Finally, in the *third* step, we propose a strategy to mitigate the bias in LLMs concerning these attributes and examine the effectiveness of our strategy in alleviating these biases. We explain each step in the rest of this section.

A. Attribute Selection

We aim to discover different attributes of the language in a labeled dataset that are potentially challenging for the model. To this end, we first use Linguistic Inquiry and Word Count (LIWC) software [28], a popular psychometric analysis tool to capture the different behavioral and topical features present in the textual messages of the dataset. These features help capture various social, cognitive, and affective processes in spam messages, supported by the literature showing the diverse behavioral tactics for deception employed by malicious actors [2], [9], [14]. These features become our candidate attributes (e.g., sad, feel, anxiety). Next, we leverage a two-sample statistical test called Kolmogorov-Smirnov (K-S) test, to investigate the significance of the attributes in distinguishing between spam and legit messages. To gather empirical evidence, we conduct K-S tests for each of the LIWC attributes across message sets of spam and legit labels in the dataset. We find statistically significant attributes ($p < 0.05$, rejecting the null hypothesis of no significance), and thereby, discover the set of remaining insignificant attributes. We argue that these attributes showing insignificance may become the sources of unintended biases in the model's prediction behavior. Thus, in our experiments, we select the insignificant attributes that are common across the datasets of different platforms to investigate the unintended bias of a model.

B. Bias Detection

To evaluate the bias of a model regarding each attribute of interest, our method includes two components: 1) *Unintended bias analysis* using quantitative metrics that are indicative of different types of biases of a model, and 2) *Attribute-based prediction analysis* for the LLMs classification decisions, by utilizing the selected metrics to uncover potential biases. We describe these two components in the following.

1) Metrics for Unintended Bias Analysis:

In order to show the presence of unintended bias in a model, we need to quantitatively measure it. While there is no consensus on what the best method to measure a bias is, we leverage the metrics studied in existing works [10], [29]. We employ the popular metric Area Under the Receiver Operating Characteristic Curve (ROC-AUC, or AUC) to measure the unintended bias. The AUC score of a classification model indicates the model's ability to separate or distinguish classes. For instance, at the highest AUC score of 1.0, the model would be perfect at predicting a spam message as spam and a legit message as legit and this could be attained across varied threshold settings, implying all legit (negative class) examples receive lower scores than all spam (positive class) examples. A common approach to studying unintended bias is to split up the test data across some groups and then, compute the bias metrics for each group. For our analysis, it is appropriate to divide a dataset into attribute-value-based groupings of 'subgroup' set and the remaining as the 'background' set. This way, we can draw comparisons between the 'subgroup' and the 'background' while only focusing on a particular attribute expressed by the subgroup.

As shown in Fig. 6, dataset is split into four subsets: negative (legit) examples in the background, positive (spam) examples in the background, negative examples in the subgroup, and positive examples in the subgroup. Based on these subsets, the three metrics (each capturing a unique aspect of the model performance), from existing literature [29], are:

Definition 1: Let D^- be the negative (legit) examples in the background set, D^+ be the positive (spam) examples in the background set, D_g^- be the negative examples in the attribute subgroup, and D_g^+ be the positive examples in the attribute subgroup.

$$Subgroup\ AUC = AUC(D_g^- + D_g^+) \tag{1}$$

$$BPSN\ AUC = AUC(D^+ + D_g^-) \tag{2}$$

$$BNSP\ AUC = AUC(D^- + D_g^+) \tag{3}$$

a) ***Subgroup AUC:*** Equation 1 gives a measure of the AUC score only from the subgroup. A higher AUC score will prove the model's understanding and distinguishability for spam and legit messages within the subgroup itself.

b) ***Background Positive Subgroup Negative (BPSN):*** Equation 2 gives a measure of the AUC score on positive examples from the background and negative examples from the subgroup. A low score here would indicate an overlap between positive examples from the background and negative examples from the subgroup. This implies that the negative examples in the subgroup have higher scores than the positive examples in the subgroup. These examples form the false positive group.

c) ***Background Negative Subgroup Positive (BNSP):*** Equation 3 gives a measure of the AUC score on the negative examples from the background and the positive examples from the subgroup. A low score here would indicate an overlap between negative examples from the background and positive examples from the subgroup. This implies that the scores in positive examples in the subgroup are lower than the score in negative examples in the subgroup. These examples form the false negative group.

*d) **Average Equality Gap (AEG) metrics**:* Further, we employ two additional metrics obtained from the Equality Gap metric proposed by [29]. The Equality Gap metric measures the difference between the true positive rates of the subgroup $(TPR(D_g))$ and the background $(TPR(D))$ given a threshold value. Equations 4 and 5 demonstrate the formula for the positive outcome of AEG and the negative outcome of AEG respectively where MWU is the Mann-Whitney U test statistic and the other variables hold from Definition 1.

$$Positive\ AEG = \frac{1}{2} - \frac{MWU(D_g^+, D^+)}{|D_g^+||D^+|} \tag{4}$$

$$Negative\ AEG = \frac{1}{2} - \frac{MWU(D_g^-, D^-)}{|D_g^-||D^-|} \tag{5}$$

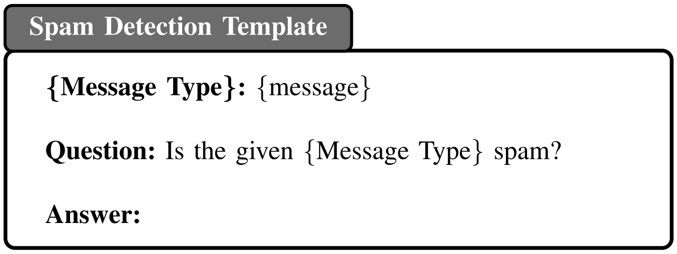
The range of AEGs is from -0.5 to 0.5, with the optimal value being 0 demonstrating that the subgroup and background distributions have identical means and no difference exists.

2) Attribute-based Prediction Analysis:

While LLMs are widely used for various classification tasks, evaluating their performance and quantifying their confidence in the result is a key area of current research. Since these models are inherently designed for text generation, when utilising them for a classification task, as studied in this paper, we need to quantify the confidence of the model about its prediction. In this work, we adopt the popular method of Predictive Entropy (PE) [30] from the literature, for calculating the LLM's confidence score for each prediction. To achieve this goal, our procedure is the following:

(i) We use the template in Fig. 2 to query the LLM on our spam detection task by providing the given input *message* and substituting *Message Type* with a conventional name for messages (i.e., *SMS* for phone messaging dataset and *Comment* for YouTube comment dataset). We prompt the LLM with this template and set a low temperature value. By selecting a low temperature value, we can consider the response of the LLM as the possible answer.

(ii) To calculate the confidence of the model regarding the possible answer, we first repeat step (i) with higher temperature value several times, which, in turn, helps create a list of brainstormed answers. In particular, higher temperature causes the model to generate different answers for the given prompt. If the model is confident about the possible answer, the brainstormed answers would align with the possible answer. Then, we utilize the template in Fig. 3 to prompt the LLM to decide on the alignment of the possible answer, by selecting *A) True*, or *B) False*, as recommended in [30].



Spam Detection Template

{Message Type}: {message}

Question: Is the given {Message Type} spam?

Answer:

Fig. 2: Prompt template for the zero-shot spam classification task.

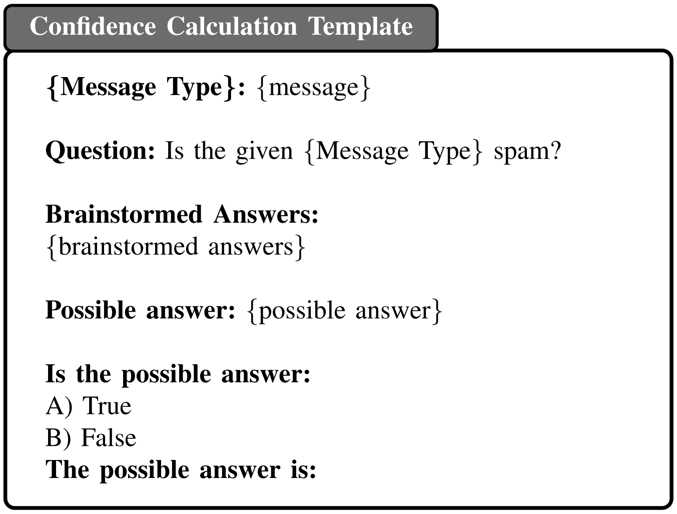
(iii) Finally, based on the PE method [30], we calculate the probability of predicting the token *A* as the next token by the model as p(True). This measure represents how much the LLM is confident about the possible answer as the prediction. Based on our task, we need to quantify the LLM's prediction and calculate p(Yes) which is the probability of predicting the given message as spam. Therefore, we measure p(Yes) based on the possible answer predicted by the model and p(True), as:

$$p(\text{Yes}) = \begin{cases} p(\text{True}), & \text{Possible Answer = Yes} \\ 1 - p(\text{True}), & \text{Possible Answer = No} \end{cases} \tag{6}$$

In our bias detection analysis, we use p(Yes) to calculate the previously discussed bias metrics and represent the distribution of the model's prediction probability. Also, to evaluate the bias of the model regarding each selected attribute, we categorize a dataset into two classes: 1) *Subgroup*: the examples for which the LIWC analysis resulted in a value greater than zero for that attribute, and 2) *Background*: those examples with zero value for the target attribute. In this way, calculating the bias metrics for a subgroup in comparison to the background can highlight the bias of the model on that attribute.

C. Bias Mitigation

Based on our preliminary experiments, we observed notable biases exhibited by the model with regard to certain behavioral attributes such as sadness and anxiety. We hypothesize that the LLMs focus more on the linguistic aspect of the text



Confidence Calculation Template

{Message Type}: {message}

Question: Is the given {Message Type} spam?

Brainstormed Answers:
{brainstormed answers}

Possible answer: {possible answer}

Is the possible answer:
A) True
B) False
The possible answer is:

Fig. 3: Prompt template for calculating the confidence of the possible answer based on PE method [30].

in the classification task and pay less attention to the socio-psychological aspects of the message’s text (e.g. the author’s goal), which are important in the presence of behavioral attributes. Therefore, we believe that augmenting the prompt by adding a hint that points to the goal of the user will help the model with a better understanding of the given message’s text. Based on this hypothesis, we propose our two-step method for mitigating the biases toward target attributes:

(i) First, for a given message, we leverage the template in Fig. 4 and prompt the LLM to analyze the message and generate the goal of the message author.

(ii) Next, we use the generated goal to enrich the prompt we use for spam classification task and help the model to take the user’s goal into consideration in its inference stage. To this end, we use the template in Fig. 5 in our bias detection pipeline for generating possible answers and brainstormed answers by the model.

Goal Generation Template

{Message Type}: {message}

Question: What is the goal of the user in the given {Message Type}?

Answer:

Fig. 4: Prompt template for generating the goal of the user for a given message.

Bias Mitigation Template

{Message Type}: {message}

Question: Is the given {Message Type} spam?

Hint: {generated goal}

Answer:

Fig. 5: Prompt template for LLM to mitigate the bias in spam classification task by incorporating the user’s goal as a hint.

IV. Experimental Setup

In this section, we describe the details of our experiments for exploring the potential bias of a spam classification model, including the datasets and experimental parameters.

A. Datsets

We conduct experiments on two different datasets to address the potential differences between different platforms through which the spam messages are sent. We analyze a dataset of YouTube comments [31] and a dataset of SMS messages [32] in our experiments. Table I shows the statistics of the datasets.

TABLE I: Distribution of data across platforms of YouTube and SMS

Dataset	# Spam	# non-Spam
YouTube	1005	951
SMS	747	4825

B. Modeling Schemes and evaluation

In this work, we evaluate the performance of the Llama3 model [33], as a state-of-the-art instruction-following LLM, on the zero-shot spam classification task to reveal the intrinsic biases regarding various attributes of the language of messages. In the bias detection step, we instruct the model with the template *“Analyze the given {Message Type} and answer the following question as briefly as possible.”* as the *system* role and the spam detection template represented in Fig. 2 as the *user* role. The former specifies the system instruction and the latter prompts the model on the spam detection task. In both templates, we use corresponding *Message Type* from {*SMS, Comment*} based on the type of message in the given dataset. In our experiments, we follow the setting in [34] and set the temperature parameter of the model to 0.001 for generating the *possible answer* to reflect the almost deterministic answer of the model and to 1.0 for generating other *brainstormed answers* to let the model explore and predict more creatively. In both cases, we limit the maximum number of tokens in the response to 10, as we expect a brief answer for the prediction. Here, we generate 10 *brainstormed answers* for each sample and utilize the confidence calculation template (Fig. 3) to calculate the confidence of the model for the predicted answer. In our implementation, we leverage the method used in [34] to calculate the probability of generating the token *‘A’* of *‘A) True’* as the next token by the model in response to the prompt, to represent the confidence of the model regarding the possible answer. To figure out the predicted answer by the model, we use a simple verbalizer which checks the existence of *’yes’* or *’no’* in the generated *possible answer*. Following that, p(Yes) is calculated based on Equation 6. Finally, this prediction probability p(Yes) is used in plotting the visualizations and computing our target bias metrics to evaluate the model’s bias.

In the mitigation step, for generating the goal of the message we use the template *“Analyze the given {Message Type} and answer the following question.”* for the *system* role and the goal generation template in Fig. 4 for the *user* role. In this step, we set the temperature parameter to 0.0 to generate the deterministic response of the model as the goal of the message and set the maximum tokens of the response to 200. Then, we leverage this *generated goal* in the bias mitigation template (see Fig. 5) to enrich the spam classification task prompt and mitigate the potential biases. We use the same pipeline as the bias detection to evaluate the proposed mitigation method.

Since we use non-zero values for the temperature parameter of the model, the results show slight differences. To address this challenge, we run 5 trials and report the mean and standard error. Table II reveals this low variation across different runs.

TABLE II: Bias metrics of the spam classification task on the YouTube and SMS datasets using Llama3 model. The AUC-based metrics improved or remained unchanged by our mitigation method are highlighted in **bold**.

Attribute	Method	YouTube Dataset					SMS Datset				
		Sub AUC	BNSP	BPSN	Neg AEG	Pos AEG	Sub AUC	BNSP	BPSN	Neg AEG	Pos AEG
sad	**baseline**	0.82±0.01	0.71±0.01	0.84±0.00	0.13±0.00	0.05±0.01	0.94±0.00	0.94±0.00	0.95±0.00	-0.04±0.00	0.01±0.03
	ours	**0.90±0.00**	**0.86±0.00**	0.83±0.01	0.04±0.01	-0.04±0.00	0.91±0.00	0.93±0.00	**0.96±0.00**	-0.05±0.00	0.08±0.02
feel	**baseline**	0.74±0.01	0.73±0.00	0.79±0.01	0.00±0.01	0.04±0.00	0.96±0.00	0.95±0.00	0.95±0.00	-0.01±0.00	-0.12±0.02
	ours	**0.85±0.00**	**0.86±0.00**	**0.80±0.00**	-0.02±0.00	-0.02±0.01	0.95±0.00	**0.95±0.00**	**0.96±0.00**	-0.03±0.01	-0.09±0.01
health	**baseline**	0.76±0.01	0.73±0.01	0.79±0.01	0.01±0.00	0.03±0.01	0.97±0.00	0.96±0.00	0.95±0.00	-0.01±0.00	-0.07±0.01
	ours	**0.87±0.01**	**0.86±0.00**	**0.79±0.01**	-0.05±0.01	-0.03±0.00	**0.97±0.00**	**0.97±0.00**	**0.96±0.00**	-0.05±0.00	-0.05±0.01
family	**baseline**	0.68±0.01	0.67±0.01	0.76±0.01	0.01±0.01	0.06±0.01	0.93±0.00	0.91±0.00	0.96±0.00	0.02±0.00	0.12±0.01
	ours	**0.68±0.01**	**0.75±0.01**	0.74±0.01	-0.08±0.01	0.06±0.01	0.90±0.01	0.88±0.00	**0.97±0.00**	0.03±0.00	0.19±0.01

V. Results and Discussion

Here, we present the results of our experiments and discuss key findings. First, we show the output of the attribute selection analysis to reveal the potential attributes contributing to unintended bias. Following that, we explore the results of our experiments on bias detection and mitigation.

A. Attribute Selection

After conducting the k-s test, we obtained the LIWC features which were insignificant in distinguishing between spam and legit messages. From these features, we employed a filtering step to select only the common insignificant attributes between the YouTube and the SMS datasets. This resulted in a total of 7 different attributes. These attributes were: sad, family, feel, anxiety, religion, death, and health. Out of these 7 attributes, certain attributes were sparsely distributed across spam and legit categories. Due to this reason, we opted to chose the attributes that had a considerable data distribution, resulting in a total of 4 attributes: sad, feel, healthy, and family. Table III provides examples of these selected attributes for the YouTube comments, indicating the behavioral tactics employed for social engineering attacks often [9]. Table IV provides the distribution of spam and legit messages across the selected attributes for all the datasets.

TABLE III: Examples of the selected attributes from YouTube dataset, demonstrating behavioral tactics employed to deceive.

Sad	Family	Feel	Health
I cried this song bringing back some hard memories.	There is one video on my channel about my brother.	Eminem is the greatest artist to ever touch the mic.	Our Beautiful Bella has been diagnosed with Wobbler's Syndrome.
This makes me miss the world cup	My uncle said he will stop smoking if this comment gets 500 likes.	Roar is without a doubt your best song...feel good song with a message for everyone	The little PSY is suffering Brain Tumor and only has 6 more months to live.

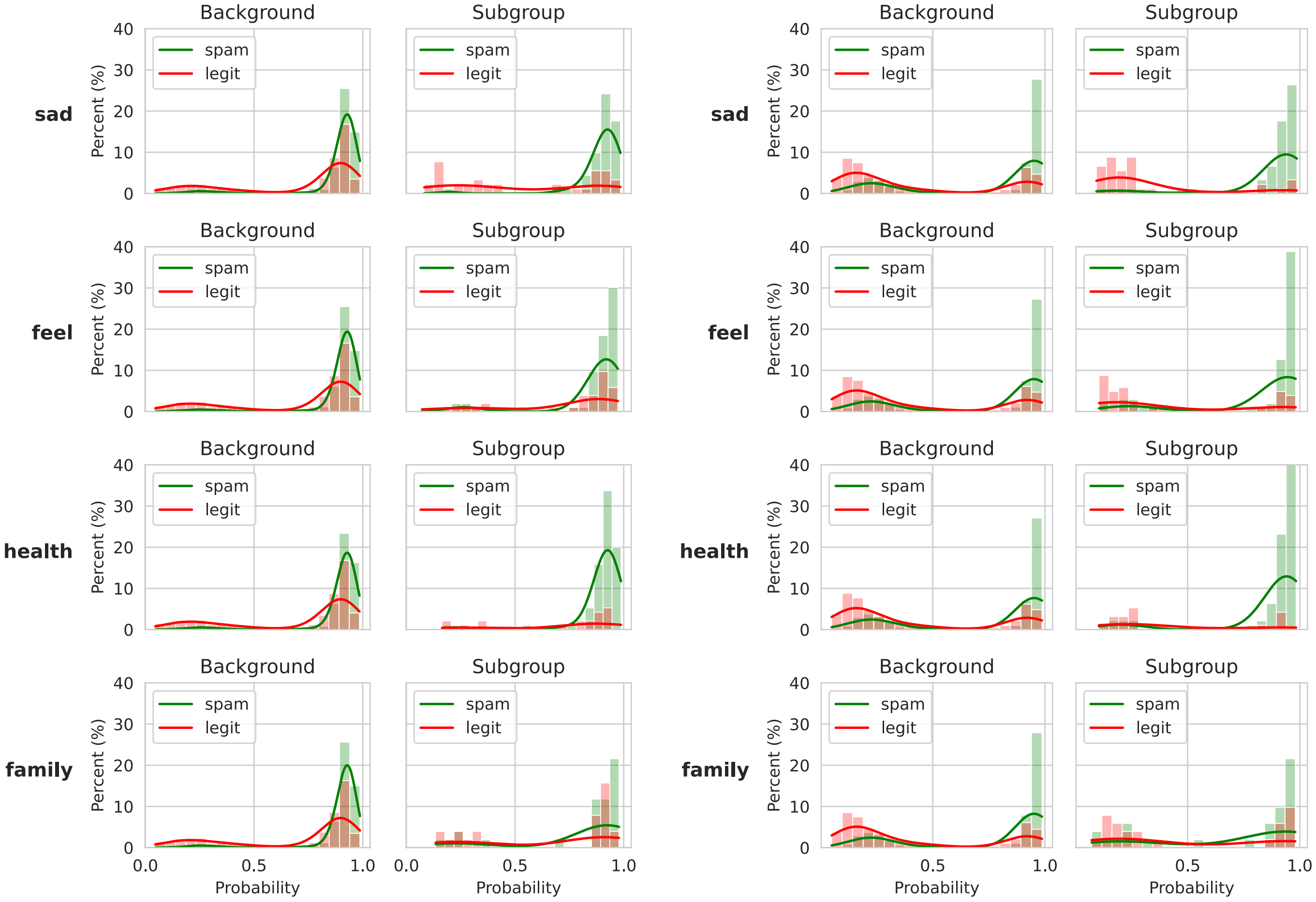
B. Bias Detection Analysis

TABLE IV: The statistics of datasets based on the target attributes.

Attribute	Category	YouTube		SMS	
		legit	**spam**	**legit**	**spam**
sad	Background	914	951	4439	724
	Subgroup	37	54	386	23
feel	Background	915	938	4514	714
	Subgroup	36	67	311	33
health	Background	932	929	4522	698
	Subgroup	19	76	303	49
family	Background	929	976	4519	737
	Subgroup	22	29	306	10

We first present a visual analysis of model predictions to demonstrate unintended biases in Fig. 6 and 9. They depict the differences in the percentage of false positives and negatives for the subgroup and background subsets of the data for a selected attribute, when considering either a baseline model or the proposed mitigation method. Further, Fig. 7a and 8a show the bias metrics for the spam detection task using Llama3 on the YouTube and SMS datasets respectively. In the YouTube dataset, there is a relatively higher BPSN and lower Subgroup AUC and BNSP value across all the attributes.

Finding 1: Lower BNSP and Subgroup AUC values reveal that the model tends to over-estimate the scores for negative samples (legit) from both the background and subgroup sets, which increases the false positive rate. The result in Fig. 6 confirms this observation, where the percentage of negative samples in higher scores is more than that in lower scores in both background and subgroup plots. Additionally, we observe that the Negative AEGs of most attributes are close to zero, that means there is not considerable shift in the distribution of scores for negative samples, except for the behavioral attribute *sad* in which there is a shift of negative samples to the right (higher scores for negative samples that can cause a higher false positive rate). In the case of Positive AEG, consistent shifts to the right for all attributes are observed, though they are marginal. These observations reveal that the model slightly tends to predict higher scores for both positive and negative samples in the subgroup were compared with those in the background. From Fig. 6, we can observe that two strong peaks have formed in the subgroup,



(a) Baseline

(b) Our mitigation method

Fig. 6: Bias mitigation results for target attributes on the YouTube dataset using Llama3 model, showing the performance gain through a mix of reduced errors for false positives and negatives.

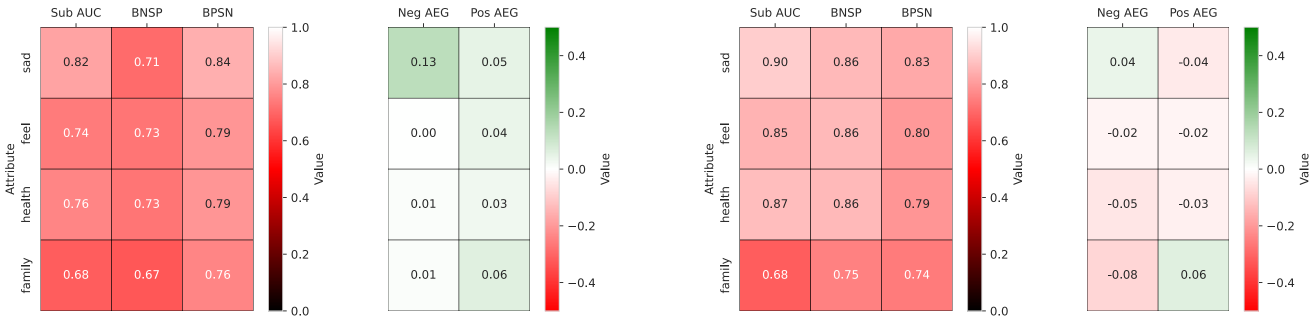
but the background only has a single peak. The first peak for spam messages has increased in the subgroup which indicates that there is an increased number of spam messages where the model is not confident, thus implying that the model is **less confident** in classifying messages as spam when they contain the above behavioral attributes in YouTube comments.

Finding 2: In all cases of the SMS dataset, the performance of the model is higher when compared with the YouTube dataset. However, the BPSN metric is higher and consistent in most cases, and Subgroup AUC and BNSP metrics are lower and inconsistent in most cases, across different attributes, similar to the YouTube dataset. Among the attributes, *sad* shows more bias in the latter two metrics. The peak of the distribution of spam messages is reduced in the subgroup as compared to the background. This indicates that there is a reduced number of spam messages where the model is strongly confident in the subgroup. Implying, that the model is confident in classifying **fewer** messages as spam when they contain the above common attributes in SMS messages. The attribute that did not show this pattern was ***health***. For this specific attribute, we saw a higher peak for spam messages in the subgroup indicating an increased number of spam messages where the model is confident and thus, implying that the model is confident in classifying **more** messages as spam when they contain the *health* attribute in the SMS Dataset.

C. Bias Mitigation Analysis

Table II shows the average of the bias metrics for 5 trials of our experiments in bias detection and mitigation steps for both datasets across different attribute settings. The bias metrics that are improved or remain stable by our proposed method are highlighted. The results reveal the effectiveness of our method in most cases. Additionally, the heat maps highlighting the results for the YouTube and SMS datasets are shown in Fig. 7b and 8b respectively.

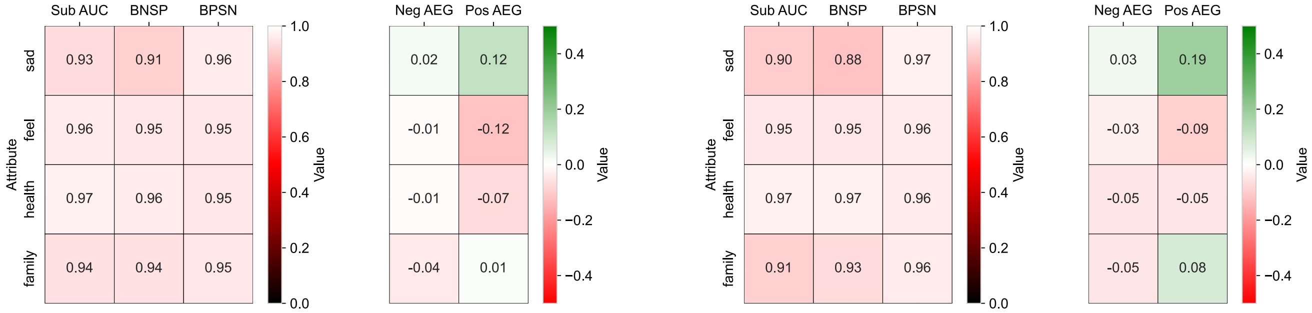
Finding 3: For the YouTube dataset, the results demonstrate the overall improvement in Subgroup AUC and BNSP in almost all cases (except for the Subgroup AUC



(a) Baseline

(b) Our mitigation method

Fig. 7: Bias metrics of the spam classification task on the **YouTube** dataset using Llama3 model.



(a) Baseline

(b) Our mitigation method

Fig. 8: Bias metrics of the spam classification task on the **SMS** dataset using Llama3 model.

of *family* attribute), while the figures for BPSN remained unchanged approximately. This observation reveals the effectiveness of our method in mitigating these two biases. These improvements are more significant for the behavioral attributes, *sad* and *feel*. Also, our method enhances these metrics for the attribute *health*. In the case of the attribute *family*, the BNSP is mitigated considerably while we can keep two other metrics stable. Regarding the effect of the proposed method on the direction of the shifts, we observe that our mitigation method generates small negative values of AEGs for both positive and negative samples in most attributes. This shift of the scores to the left is also observed in Fig. 6b. The bias detection revealed that the model tends to over-estimate the scores and the shift of the scores to the left can alleviate this bias, as the result shows the effectiveness of the proposed method in increasing the AUC-based metrics. However, since these shifts are small and none of the metrics in this paper can reflect the pattern of the shift (how this shift is distributed) we cannot explore the exact effect of the shift on the bias metrics.

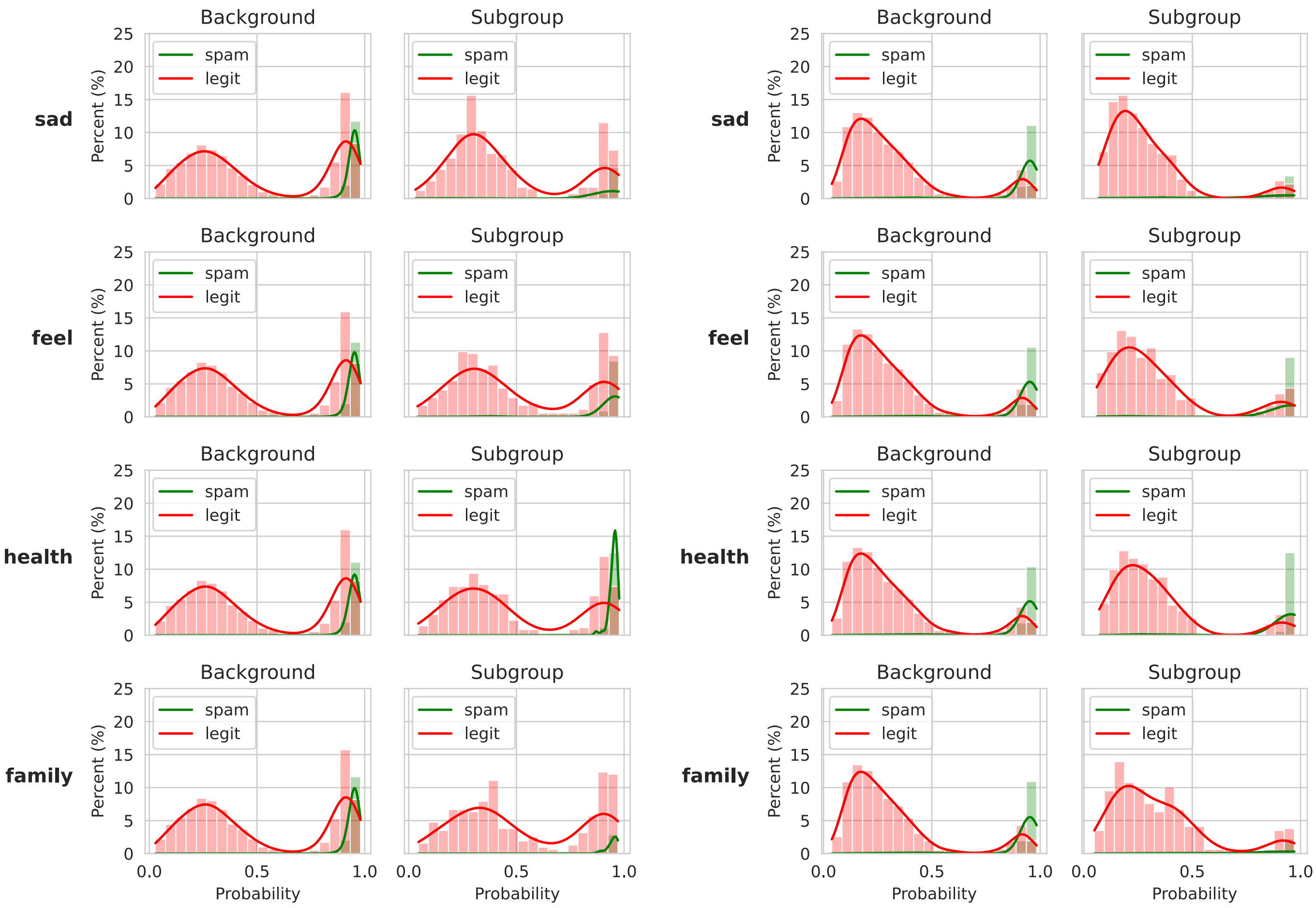
Finding 4: For the SMS dataset, BPSN is the metric that is improved in all cases, despite the high performance of the baseline. Also, for other metrics, including Subgroup AUC and BNSP, our mitigation method shows almost consistent results in *feel* and *health* attributes, although slight decreases are observed in *sad* and *family* attributes. Fig. 6 reveals a considerable shift to the left for the scores in our mitigation method, mostly for negative samples. This observation is aligned with the result of the Negative AEG metric in Fig. 8b, where they are mostly negative.

D. Error Analysis

We analyzed the dataset across the chosen attributes for the YouTube dataset and presented some comments with complex language in Table V. They show the dual nature of words and the significance of context. For example, the word **'LOST'** is actually being used as a proper noun in the context of the *sad* attribute, thus, challenging the model to catch this nuance. Similarly, the word **'sick'** here is being used as slang for cool, but the LIWC model associates this word with *health*. In the case of the *feel* attribute, we saw the word **'cool'** being used frequently with little to no context, making it more difficult for the LLM model to interpret anything meaningful.

VI. Conclusion, Limitations, and Future Work

In this paper, we explore the potential application of employing state-of-the-art LLMs in the current cyber-threat landscape. LLMs with their remarkable adaptability and existing knowledge from pre-training on enormous data, present an encouraging solution in the domain of spam detection. We employ a set of existing metrics for unintended bias detection and propose a zero-shot spam classification approach, which not only detects bias due to behavior attributes of language in



(a) Baseline

(b) Our mitigation method

Fig. 9: Bias mitigation results for target attributes on the SMS dataset using Llama3 model, showing the performance gain through a mix of reduced errors for false positives and negatives.

TABLE V: Examples of complex messages across attributes from the YouTube dataset.

Sad	Health	Feel
Charlie from LOST??	Sick Music for sick females?	This is so cool, why haven't I heard this before??
Who else saw jesses dancing sorry if I spelled it wrong peace??	this song always gives me chills! :)	Check out my youtube channel for cool beatboxing

spam content but also mitigates it effectively while leveraging LLMs. It helps reduce the cost of extensive human effort in data labeling requirements in traditional supervised learning methods, while still achieving an improved performance. The unintended bias metrics we leveraged in this work revealed nearly identical biases towards behavioral attributes of the language of spams across our datasets from SMS and YouTube platforms. Moreover, the result demonstrated the effectiveness of our proposed goal-oriented mitigation method in improving the bias metrics. Our results can lay the groundwork for availing the power of LLMs while ensuring bias detection and mitigation safeguards for inclusive cybersecurity solutions.

Limitations and Future Work: Our analysis of the results demonstrates the existing challenges present in the domain of bias detection and mitigation. This work explored only the biases of the Llama3 model on the datasets from SMS and YouTube platforms. In the future, we plan to experiment with other more diverse communication platforms such as Twitter, emails, and other messaging services, as well as evaluate the biases of other popular LLMs on the spam detection task. Our novelty is not in the prompt design as we utilized a standard prompt template from the existing literature. In the future, we hope to do a comprehensive evaluation on how different prompts can affect the bias measurement and mitigation results which can further help design more efficient prompt strategies. While our results show the effectiveness of the proposed method in several attributes, our method is limited to behavioral attributes. Future works should explore other types of attributes corresponding to various tactics of social engineering attacks, which can be the source of biases for our task. Further, different mitigation strategies need to be explored to process messages with different attributes to achieve a more effective mitigation strategy.

Acknowledgment

Authors thank the U.S. National Science Foundation for grant # 2210107 and the Commonwealth Cyber Initiative for grant # HN-4Q24-057 to partially support this research.

References

[1] U.S. Federal Trade Commission, "Consumer sentinel network data book 2023," accessed: August 10, 2024. [Online]. Available: https://www.ftc.gov/system/files/ftc_gov/pdf/CSN-Annual-Data-Book-2023.pdf.

[2] S. Rao, A. K. Verma, and T. Bhatia, "A review on social spam detection: Challenges, open issues, and future directions," *Expert Systems with Applications*, vol. 186, p. 115742, 2021.

[3] Z. B. Siddique, M. A. Khan, I. U. Din, A. Almogren, I. Mohiuddin, and S. Nazir, "Machine learning-based detection of spam emails," *Scientific Programming*, vol. 2021, no. 1, p. 6508784, 2021.

[4] T. Xia and X. Chen, "A weighted feature enhanced hidden markov model for spam sms filtering," *Neurocomputing*, vol. 444, pp. 48–58, 2021.

[5] İ. Yurtseven, S. Bagriyanik, and S. Ayvaz, "A review of spam detection in social media," in *2021 6th International Conference on Computer Science and Engineering (UBMK)*. IEEE, 2021, pp. 383–388.

[6] L. He, X. Wang, H. Chen, and G. Xu, "Online spam review detection: A survey of literature," *Human-Centric Intelligent Systems*, vol. 2, no. 1, pp. 14–30, 2022.

[7] O. Abayomi-Alli, S. Misra, A. Abayomi-Alli, and M. Odusami, "A review of soft techniques for sms spam classification: Methods, approaches and applications," *Engineering Applications of Artificial Intelligence*, vol. 86, pp. 197–212, 2019.

[8] S. B. Abkenar, M. H. Kashani, M. Akbari, and E. Mahdipour, "Learning textual features for twitter spam detection: A systematic literature review," *Expert Systems with Applications*, vol. 228, p. 120366, 2023.

[9] R. Montañez, A. Atyabi, and S. Xu, "Social engineering attacks and defenses in the physical world vs. cyberspace: a contrast study," in *Cybersecurity and Cognitive Science*. Elsevier, 2022, pp. 3–41.

[10] L. Dixon, J. Li, J. Sorensen, N. Thain, and L. Vasserman, "Measuring and mitigating unintended bias in text classification," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018, pp. 67–73.

[11] S. Caton and C. Haas, "Fairness in machine learning: A survey," *ACM Computing Surveys*, vol. 56, no. 7, pp. 1–38, 2024.

[12] A. Atabek, E. Eralp, and M. E. Gursoy, "Trust, privacy and security aspects of bias and fairness in machine learning," in *2023 5th IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*. IEEE, 2023, pp. 111–121.

[13] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 3, pp. 1–45, 2024.

[14] A. Das, S. Baki, A. El Aassal, R. Verma, and A. Dunbar, "Sok: a comprehensive reexamination of phishing research from the security perspective," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 671–708, 2019.

[15] M. F. A. Kadir, A. F. A. Abidin, M. A. Mohamed, and N. A. Hamid, "Spam detection by using machine learning based binary classifier," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 26, no. 1, pp. 310–317, 2022.

[16] T. Manyumwa, P. F. Chapita, H. Wu, and S. Ji, "Towards fighting cybercrime: Malicious url attack type detection using multiclass classification," in *2020 IEEE international conference on big data (Big Data)*. IEEE, 2020, pp. 1813–1822.

[17] A. B. Nassif, M. A. Talib, Q. Nasir, and F. M. Dakalbab, "Machine learning for anomaly detection: A systematic review," *Ieee Access*, vol. 9, pp. 78 658–78 700, 2021.

[18] N. Hussain, H. T. Mirza, I. Hussain, F. Iqbal, and I. Memon, "Spam review detection using the linguistic and spammer behavioral methods," *IEEE Access*, vol. 8, pp. 53 801–53 816, 2020.

[19] S. S. Roy, P. Thota, K. V. Naragam, and S. Nilizadeh, "From chatbots to phishbots?: Phishing scam generation in commercial large language models," in *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 2024, pp. 221–221.

[20] D. Boumber, B. E. Tuck, R. M. Verma, and F. Z. Qachfar, "Llms for explainable few-shot deception detection," in *Proceedings of the 10th ACM International Workshop on Security and Privacy Analytics*, 2024, pp. 37–47.

[21] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM computing surveys (CSUR)*, vol. 54, no. 6, pp. 1–35, 2021.

[22] M. Hort, Z. Chen, J. M. Zhang, M. Harman, and F. Sarro, "Bias mitigation for machine learning classifiers: A comprehensive survey," *ACM Journal on Responsible Computing*, vol. 1, no. 2, pp. 1–52, 2024.

[23] B. Van Giffen, D. Herhausen, and T. Fahse, "Overcoming the pitfalls and perils of algorithms: A classification of machine learning biases and mitigation methods," *Journal of Business Research*, vol. 144, pp. 93–106, 2022.

[24] Y. Hswen, A. Gopaluni, J. S. Brownstein, J. B. Hawkins *et al.*, "Using twitter to detect psychological characteristics of self-identified persons with autism spectrum disorder: a feasibility study," *JMIR mHealth and uHealth*, vol. 7, no. 2, p. e12264, 2019.

[25] L. Yu, G. Mottola, C. N. Kieffer, R. Mascio, O. Valdes, D. A. Bennett, and P. A. Boyle, "Vulnerability of older adults to government impersonation scams," *JAMA Network Open*, vol. 6, no. 9, pp. e2 335 319–e2 335 319, 2023.

[26] J. Shao, Q. Zhang, Y. Ren, X. Li, and T. Lin, "Why are older adults victims of fraud? current knowledge and prospects regarding older adults' vulnerability to fraud," *Journal of elder abuse & neglect*, vol. 31, no. 3, pp. 225–243, 2019.

[27] G. Gressel, R. Pankajakshan, and Y. Mirsky, "Discussion paper: Exploiting llms for scam automation: A looming threat," in *Proceedings of the 3rd ACM Workshop on the Security Implications of Deepfakes and Cheapfakes*, 2024, pp. 20–24.

[28] Y. R. Tausczik and J. W. Pennebaker, "The psychological meaning of words: Liwc and computerized text analysis methods," *Journal of language and social psychology*, vol. 29, no. 1, pp. 24–54, 2010.

[29] D. Borkan, L. Dixon, J. Sorensen, N. Thain, and L. Vasserman, "Nuanced metrics for measuring unintended bias with real data for text classification," in *Companion proceedings of the 2019 world wide web conference*, 2019, pp. 491–500.

[30] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson *et al.*, "Language models (mostly) know what they know," *arXiv preprint arXiv:2207.05221*, 2022.

[31] T. Alberto and J. Lochter, "YouTube Spam Collection," UCI Machine Learning Repository, 2017, DOI: https://doi.org/10.24432/C58885.

[32] T. Almeida and J. Hidalgo, "SMS Spam Collection," UCI Machine Learning Repository, 2012, DOI: https://doi.org/10.24432/C5CC84.

[33] AI@Meta, "Llama 3 model card," 2024. [Online]. Available: https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md

[34] L. Kuhn, Y. Gal, and S. Farquhar, "Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=VD-AYtP0dve