Using Transformers for Identification of Persuasion Principles in Phishing Emails

Bimal Karki[1], Faranak Abri[2], Akbar Siami Namin[1], Keith S. Jones[3]
[1, 2]Department of Computer Science, [3]Department of Psychological Sciences
[1,3]Texas Tech University, [2]San Jose State University
{*bimal.karki, akbar.namin, keith.s.jones*}*@ttu.edu* | *faranak.abri@sjsu.edu*

2022 IEEE International Conference on Big Data (Big Data) | 978-1-6654-8045-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/BigData55660.2022.10020452

***Abstract*—It is important to learn about attackers and their attacking strategies so that better and more effective defense systems can be built. During the reconnaissance stage, attackers intend to probe potential targets through various techniques including social engineering attacks. Phishing through email is a well-known, cheap, easy, and surprisingly effective technique for obtaining the needed information. This type of attack targets individuals and thus utilizes weaknesses that might exist in each person. Given the uniqueness of each individual's personality, attackers make sure the right persuasion principle technique is employed for each targeted individual. This paper describes efforts to build machine-learning transformers, the emerging technique in language modeling, with the goal of building classifiers that take into account different types of persuasion principles. More specifically, the paper describes efforts to build machine-learning transformers based on BERT, RoBERTa, and DistilBERT and captures their classification results. The results show that these transformers are accurate enough to build a classification of phishing emails with respect to persuasion techniques. Furthermore, we report that the RoBERTa model is able to train faster than BERT and DistilBERT models.**

***Index Terms*—Persuasion principles, machine-learning transformers, phishing attacks.**

I. Introduction

Modern communication depends heavily on emails. Email is an official form of communication in almost all organizations. With the rise in use of emails, crimes related to this communication channel have also increased. One of the forms of misuse of email is phishing attacks. Phishing is a type of social engineering during which an attacker sends fake or false information to trick users to click on a link or respond to the received email so these victims expose their sensitive and private information. Phishing is seen as a modern way of stealing. Various techniques are used to formulate and detect phishing emails such as word-embedding [1]. One of the ways to formulate an effective phishing email is to manipulate the psychological state of mind of users. By carefully manipulating the psychological, emotional, and state of mind of individuals such as fear and greed, an attacker will be able to obtain various information or even money from their targets.

Cialdini [2] identified six principles of persuasion tactics. These tactics utilize various psychological states of individuals to lure them to do something on behalf of the attacker. It is important to understand attacker's strategies so a better defense mechanism can be adapted. Based on Cialdini's persuasion principles [2], there are six basic tendencies of human behavior that can generate a positive response: 1) *Reciprocation*, 2) *Consistency*, 3) *Social Proof*, 4) *Likeability*, 5) *Authority*, and 6) *Scarcity*. Phishers use these techniques to a great extent to persuade users to share their credentials, send money to them, do unsolicited financial transactions, or do illegal things. Our goal is to find out which of the persuasion principles are used more in phishing emails. The existing spam detection techniques and models mostly formulate the problem as a binary classification problem (i.e., categorizing emails as phishing or not). We want to go a step further and find out what kind of persuasion tactics are used in phishing emails through transformers, an emerging language modeling technique in Natural Language Processing (NLP).

This paper focuses on the problem of the classification of emails with respect to persuasion principles. More specifically, the objective is to find out whether these principles are utilized in crafting phishing emails and whether it is possible to automatically detect them using classification techniques in Natural Language Processing (NLP). In this paper, we explore different techniques and in particular machine learning-based transformers including BERT [3], RoBERTa [4] and DistilBERT [5] to classify emails into categories defined according to Cialdini's principles [2].

We also utilize Latent Dirichlet Allocation (LDA) for automated topic modeling that can enable labeling of the given emails. The utilization of LDA in labeling enables us to formulate the problem as a supervised learning problem and thus be able to categorize emails with respect to persuasion principles. Heijden and Allodi [6] also used LDA for the same purpose. Our experiment shows that LDA is not very efficient or effective in classifying emails. The topic modeling offered by LDA was too broad and vague, so it could not offer any meaningful classification result. On the other hand, the "*similarity matching*" [7], also known as "*semantic similarity*", of transformer-based BERT, RoBERTa, and DistilBERT models were able to classify emails with reasonable accuracy. This paper makes the following key contributions:

1) Introduce a methodology for labeling unlabeled data using LDA and BERT similarity measurement,
2) Demonstrate that the use of LDA for topic modeling and performing classification for persuasion principles produces imprecise prediction models.
3) Demonstrate that, unlike LDA, the transformer-based

classifiers and similarity matching offered by transformers produces good results for the classification problem stated in this work,
4) Demonstrate that RoBERTa converges faster than BERT and DistilBERT to produce the classification results.

The rest of this paper is organized as follows: Section II reviews the related work on this subject. The technical background is briefly presented in Section III. Section IV presents the methodology, architecture of the models, and the algorithms developed in this work. The experimental setup, including the dataset studied and its pre-processing, is discussed in Section V. The results of the work are presented and discussed in Section VI. The threats to validity of the experiments conducted in this paper are discussed in Section VII. Section VIII concludes the paper and highlights the potential future research directions.

II. Related Work

Spam and phishing email detection has become one of the major security problems. A considerable amount of research has investigated spam and phishing email detection [1], but very little research has been done on the categorization of phishing emails based on Cialdini's persuasion principles. Most of the previous and current research focuses on classifying emails into phishing and legit emails. However, Akbar [8] developed a flowchart to identify if emails incorporated Cialdini's persuasion principles. According to Akbar, 96.1% of the emails tested utilized some kind of *authoritative* language; whereas, 41.1% of the emails tested employed language related to *scarcity*.

Heijden and Allodi [6] discussed a quantitative approach to measure the cognitive vulnerability in phishing emails with the goal of estimating the success rate of phishing email campaigns. The techniques they presented rely on the use of Labelled Latent Dirichlet Allocation (LLDA) for enabling supervised topic modeling. In this paper, we adapted a similar task, but in addition to using LLDA, we also utilized machine-learning transformers to 1) similarity match emails for the purpose of labeling, and 2) perform classifications on the synthesized dataset.

Nishikawa et al. [9] proposed a method to extract the persuasion method that an attacker employed in their emails using logistic regression and neural networks. In their research, the authors studied the Enron dataset [10], which is a dataset of benign emails rather than a repository of phishing emails.

Stojnic et al. [11] used word frequency and word occurrence techniques to classify emails. More specifically, the authors used natural language processing techniques, such as TF-IDF weighting, k-Means clustering, and LDA topic modeling, to analyze the persuasion techniques employed by attackers.

Other approaches in this line of research such as CipherCraft/Mail [12] inspect behavioral factors, including the sender's domain authentication, icons, and images in emails. Sevtap et al. [3] proposed a technique to detect a malicious email by looking for a specific feature.

The work presented in this paper is close to the experiment reported by Heijden and Allodi [6]. Similar to Heijden and Allodi [6], we also utilized LDA to perform topic modeling and thus label phishing emails with no success. However, unlike Heijden and Allodi [6], we employed machine-learning transformers, the cutting-edge approaches in language modeling, to build classifiers for classifying phishing emails based on persuasion principles.

III. Technical Background

This section briefly introduces the theoretical background of the machine learning approaches adapted in this research work.

A. Latent Dirichlet Allocation (LDA)

LDA is an unsupervised learning method that classifies texts into groups and particular topics. It builds a topic-per-document model and word-per-topic model, which is modeled as a Dirichlet distribution. This process is a distribution over distributions, meaning that each draw from a Dirichlet process is itself a distribution. What this implies is that a Dirichlet process is a probability distribution wherein the range of this distribution is itself a set of probability distributions. LDA works on two major assumptions:

1) Documents contain a mixture of topics, and
2) Topics are a mixture of tokens.

The first step is to clean, pre-process, and tokenize the given text. The cleaning process includes removing stop words, removing words that do not help in generating topics, and removing other language-relevant words. For pre-processing, we Lemmatization and stemmanize the words and tokenize them.

According to LDA, every word is associated (or related) with a latent (i.e., hidden) topic. The latent assignment of a topic word in the documents gives a topic word distribution present in the corpus. LDA uses an iterative process over the document to find the best distribution of the words and topics.

B. The BERT Transformer Model

BERT stands for Bidirectional Encoder Representations from Transformers. BERT is based on Transformers, a deep learning model in which every output element is connected to every input element, and the weightings between them are dynamically calculated based on their connections. Historically, language models could only read text input sequentially – either left-to-right or right-to-left – but could not do both at the same time, or more specifically, at one time. Most language models can read the input either right to left or left to right but BERT is different because it can read the input data in both directions at once. This capability is called bi-directionality and this is possible because of the transformers.

Transformers were first introduced by Google in 2017. The most popular techniques for solving NLP-based problems were through using conventional approaches such as Convolutional Neural Networks (CNN), Recursive Neural Networks (RNN), and Long-Short Term Memory (LSTM). Because these models

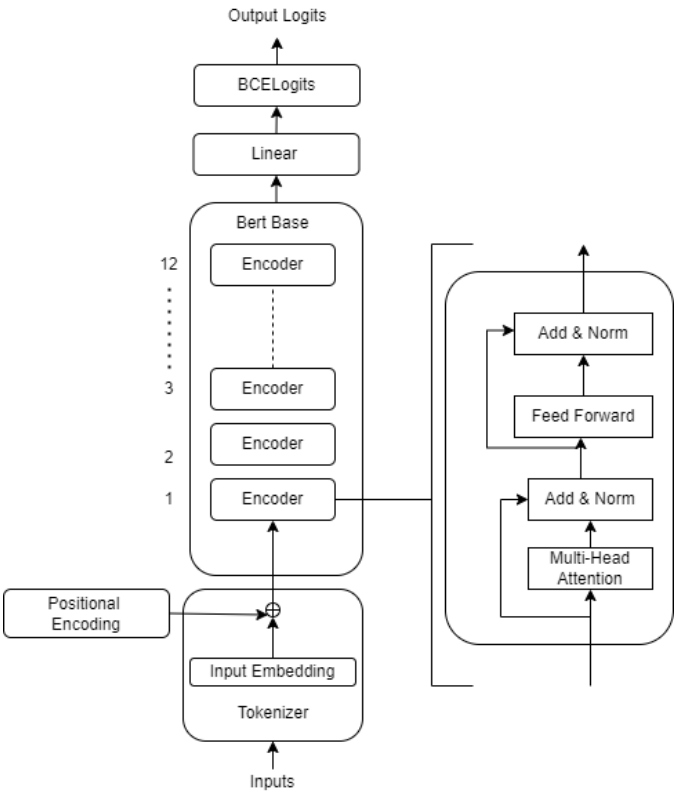


Fig. 1: The adapted architecture of BERT.

could only read input sequences from either direction (left to right and vice versa), after a certain point the information in the input was lost and the true meaning of the paragraph could not be preserved. Transformers, in general, and BERT, in particular, address this problem.

BERT has been trained with unlabeled data, plain text corpus, namely the English Wikipedia, and a Brown corpus. The pre-trained model serves as a base model and the model can be modified to do various tasks as needed. An advantage of BERT is that the model does not need to be retrained; it can just be fine-tuned. This technique is known as transfer learning. By adding a few layers to the base model, one can tune this model for a specific task. As a result, the user of the BERT model does not necessarily need sophisticated hardware or a very large dataset. Furthermore, the training time for fine-tuning is much lower. In our classification experiment reported in this paper, we added "*a classification layer*" to the BERT model to fine-tune the model and achieve our goal of accurate classification.

C. The RoBERTa and DistilBERT Transformer Models

RoBERTa is a Robustly Optimized BERT Pretraining approach, developed by Facebook AI. It improves on the popular BERT model by modifying key hyperparameters and pre-training on a larger corpus. This leads to improved performance compared to the basic BERT model. RoBERTa is more lightweight than the BERT model.

DistilBERT is a small, fast, and lightweight model developed based on the BERT model. DistilBERT can be trained faster with a smaller dataset. It has 40% less parameters and can train 60% faster while preserving 95% of the BERT's performance [13].

In this paper, we adapted the RoBERTa and DistilBERT models to compare our results with the BERT model. Because these models are more optimized versions of BERT, we hoped to achieve a better result with these models.

IV. Architecture & Algorithm

A. The Architecture

The architecture is similar to the basic form of BERT models and thus contains all the components of BERT since we fine-tuned the models. The key difference is that we added a linear classification layer connecting the last hidden layer of BERT to give output (i.e., five classification topics or persuasion principles). We used `BCEWithLogitsLoss` as an activation function. This function combines the sigmoid activation function and BCELoss function into one single component. Figure 1 shows the adapted BERT architecture implemented and studied in this work.

The models take sentences and labels and tokenize them. The labels are generated automatically (discussed in Section V-C). The tokenized terms are then vectorized using word embedding and thus mapped to numbers. Given that sentences are of different lengths, we padded the tokenization. The tokenizer returns "input_ids", "attention_masks" and "labels" tensor. This tensor is fed into each model. We did not freeze the layers of BERT models meaning that all data pass through every layer. The output of the model are logit values, which can be converted to probability values. However, instead of utilizing these probability values, we employed a threshold value to decide about classification and its accuracy directly.

B. The Algorithm

The machine learning pipeline for building, fitting, and testing the transformer-based models is given in Algorithm 1. The pipeline starts off with accepting unlabeled data and a threshold value to adjust the error rates of the classification performed by each transformer.

The unlabeled data then are labeled by the procedure explained in Section V-C. The labeled data are then divided into training, validation, and testing subsets. Once the transformer-based classification models are built, they are tested and the accuracy is adjusted with the threshold value with the goal of optimizing the classification accuracy.

V. Experimental Setup

A. Dataset

The dataset used in this work is a set of phishing emails obtained online [14]. The dataset contains $18,819$ phishing emails. In this paper, the emails in this dataset are first labeled using transformer semantic matching where the labels were drawn from the persuasion principles: 1) Reciprocity, 2) Commitment, 3) Liking, 4) Scarcity, 5) Authority, 6) Social

TABLE I: The initial set and the number of similar emails matched using BERT similarity matching.

Persuasion	Initial Set	# Emails
Reciprocity	– Please follow the link below to update the password. – to continue using our service please download the file attached to this email and update your login information. – This email has been checked for viruses by Avast antivirus software. – We need you to update your information for further use of your PayPal account. – In return we will credit $20 to your account - Just for your time!	511
Authority	– They're not Shopify employees they're designers, developers and consultants chosen for their deep knowledge and successful track record. – Tried to access your personal account! please click the link below and enter your account information to confirm that you are not currently away. – Update your account dear valued customer we regret to inform you that your account at eBay could be suspended if you don't update your billing information. – We recently contacted you after noticing an issue on your account.	953
Scarcity	– Booked 2 times for your dates in the last 24 hours on our site	455
Social Proof	– Speak to an apple expert now get your questions answered by an expert via phone, chat, email, or even twitter.	1464
Commitment	– From our buyer and seller protection policies to our verification and reputation systems, we'll help to keep you safe. – PayPal is committed to maintaining a safe environment for its community of buyers and sellers. – To protect the security of your account, PayPal employs some of the most advanced security systems in the world and our anti-fraud teams.	541
Total		3,924

Algorithm 1 The machine learning pipeline.

Inputs:
unlabeled data
A threshold value

Output:
precision, recall, and F1 scores

Generate label for input data
$Labeled\_Data \leftarrow$ labeled input data
Split $Labeled\_Data$ into (train, validation, testing)
Build Transformer Models:
  $Model_{BERT}(train, validation)$
  $Model_{RoBERTa}(train, validation)$
  $Model_{DistilBERT}(train, validation)$
Test each model:
  $Model_{BERT}(train, validation)$
  $Model_{RoBERTa}(train, validation)$
  $Model_{DistilBERT}(train, validation)$
if predicted category $>$ threshold **then**
  category $\leftarrow$ highest prediction value
else
  No category
end if
return Report precision, recall, and F1 for each model

Proof, 7) Consistency, and 8) Other. Then the labeled emails are fed into three transformers to build classification models.

B. Pre-processing

The phishing dataset is an unlabeled dataset. We developed a method (Section V-C) to label each email with respect to Cialdini's principles. After applying our procedure, the number of emails was reduced by a great number. This is because not all emails follow or contain the principles of persuasion. Emails contain general sentences like greetings, introductions and so on that may not be relevant to any persuasion principles. As a result, we employed the threshold value in Algorithm 1 to discard emails whose threshold values were below a certain value ($= 0.6$; more specifically, $similarity\_sentence() > 0.6$).

TABLE II: The number of balanced samples for each category (R:Reciprocity; C:Commitment; S:Scarcity; A:Authority; SP:Social Proof).

	#R	#C	#S	#A	#SP	#Total
Unbalanced	511	541	455	953	1,464	3,924
Balanced	1,022	1,082	910	1,250	1,146	5,410

The final number of labeled emails was $3,065$ with five persuasion principles 1) Reciprocity, 2) Commitment, 3) Scarcity, 4) Authority, and 5) Social Proof.

C. Labelling the Input Data (Emails)

LDA is an unsupervised technique and therefore the data do not need to be labelled. However, for training transformer-based models, such as BERT, RoBERTa, and DistilBERT, the data need to be labeled. One of the main challenges in this field of study is the lack of labeled datasets. Moreover, to build a good machine-learning transformer model, we need a huge dataset to train and it is infeasible to manually label each email due to the tagging cost and lack of accuracy involved in human error.

We developed a method to automatically generate a label for each sample in the unlabeled dataset. We used the sentence matching (i.e., semantic similarity) method to generate this labeled dataset. First, we manually tailored (i.e., synthesized) a few emails, 5 – 6 sample sentences/emails in our case, for each topic (i.e., principle). We then used sentence matching (i.e., similarity matching or semantic similarity) offered by BERT [15] methods to categorize the sentences in phishing emails and generate a labeled dataset. Table I lists the initial sets of email samples synthesized for each category or principle along with the total number of phishing emails matched (i.e., using similarity matching) to each category.

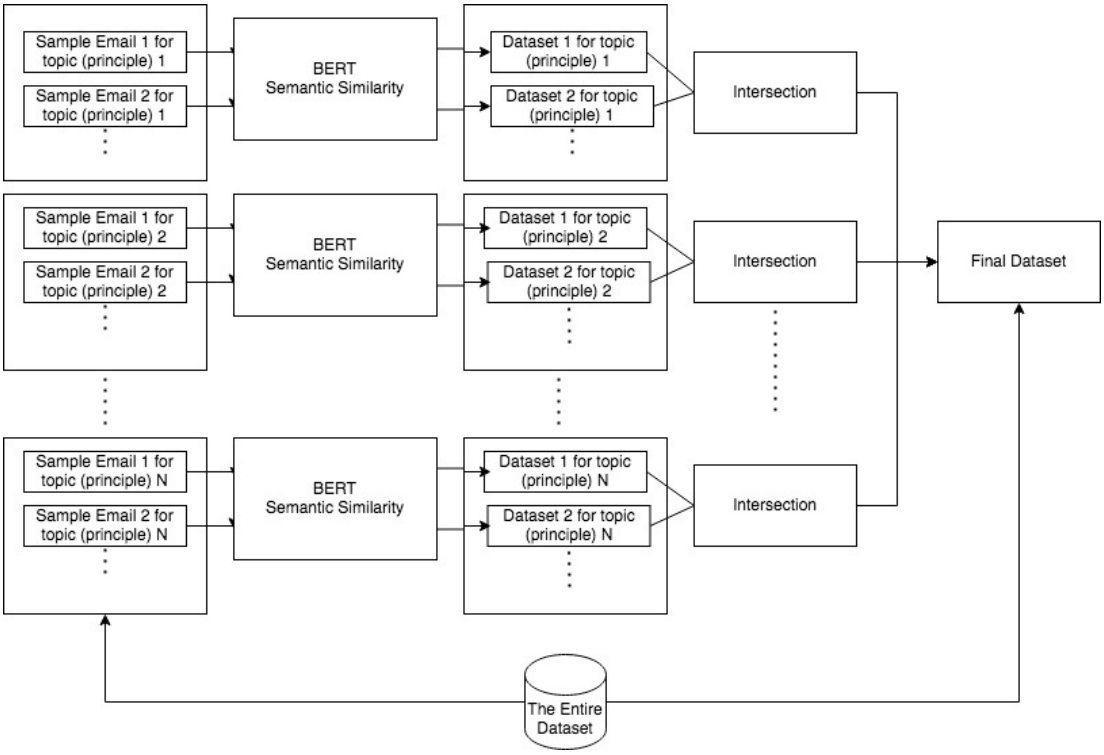


Fig. 2: Preparation of labeled data.

TABLE III: LDA - 5 Topics.

Topic	Words (Similarity Value)
0	password(0.028), mail(0.028), address(0.027), click(0.025), link(0.025), protect(0.021), assistance(0.020), page(0.020), help(0.019), reply(0.018)
1	information(0.042), update(0.032), online(0.018), service(0.018), record(0.017), click(0.015), dear(0.014), user(0.010), privacy(0.010), link(0.010)
2	access(0.031), message(0.021), item(0.021), policy(0.019), security(0.018), protect(0.015), help(0.015), limited(0.014), send(0.013), information(0.012)
3	purchase(0.011), good(0.010), size(0.008), today(0.008), table(0.008), face(0.008), track(0.007), width(0.007), shipping(0.006), sell(0.006)
4	customer(0.021), credit(0.017), service(0.015), contact(0.010), make(0.010), security(0.009), online(0.009), part(0.009), time(0.009), secure(0.009)

As indicated in Table I, the majority of phishing emails utilize "social proof" as their persuasion principle (1464 phishing emails), followed by authority with 953 phishing emails. This outcomes are somewhat consistent with the results reported by Akbar [8] where it is reported that authority is the most popular persuasion technique in phishing emails followed by scarcity, consistency, and likeablity. It is also consistent with the results reported by Heijden and Allodi [6] where it is reported that the Authority, Scarcity and Liking principles are the most popular persuasion techniques employed in phishing attacks. Furthermore, this observation is also consistent with the results reported for vishing [16] in employing persuasion principles where they report a majority of social engineering attacks employ social proof and authority as their primary techniques in launching social engineering attacks.

For each manually tailored sentence, one dataset is synthesized. Next, we take an intersection of the sentences with a certain threshold value to build the final labeled dataset for each category (i.e., principle). We then merged these final datasets for each category to obtain the final labeled dataset. Figure 2 illustrates the process of building the final dataset.

An important note in building a fair machine-learning model is that it should be trained on a balanced dataset. Therefore, while merging the datasets, the number of samples in each category should be balanced. In our experiment, we noticed that "authority" and "social proof" have more matched patterns (i.e., sentences) than other topics. We employed over/under sampling techniques to make the dataset balanced. Table II lists the number of unbalanced and then balanced samples in the dataset.

At first, we applied the Genism (`corpora()`) [17] and SpaCy (`nlp(), doc.similarity()`) [18] libraries for sentence matching. The method used by these libraries to match sentences did not yield good results. These libraries looked at the structure of the sentences (i.e., syntax) more than the meaning and semantics of the sentence (i.e., semantic). As a result, we used the word-embedding offered by the BERT sentence matching [15], which looks at syntactic and, to an extent, the semantic meaning of the sentences.

TABLE IV: LDA - 25 Topics.

Topic	Words (Similarity Value)
0	chase(0.056), survey(0.038), renew(0.023), bank(0.022), online(0.021), service(0.016), jpmorgan(0.015), customer(0.014), reward(0.012), time(0.010)
1	well(0.058), fargo(0.057), citibank(0.036), suspect(0.017), appreciate(0.017), bank(0.017), integrity(0.017), online(0.016), network(0.015), unauthorized(0.015)
2	para(0.158), cuenta(0.061), conta(0.056), cliente(0.052), sistema(0.048), este(0.044), banco(0.044), nosso(0.044), todos(0.043), mail(0.032)
3	quota(0.103), increase(0.042), mailbox(0.040), webmail(0.029), hide(0.027), limit(0.026), size(0.026), folder(0.023), mail(0.022), validate(0.022)
4	sorry(0.067), mailbox(0.001), validate(0.001), administrator(0.001), mail(0.001), bank(0.001), exceed(0.001), password(0.001), quota(0.001), limit(0.001)
5	verify(0.018), visa(0.018), download(0.017), card(0.015), attempt(0.014), choose(0.014), security(0.013), initiate(0.013), attachment(0.012), recently(0.012)
6	webmail(0.036), mail(0.025), microsoft(0.024), upgrade(0.024), mailbox(0.022), database(0.022), paste(0.021), administrator(0.020), browser(0.019), maintenance(0.019)
7	access(0.048), limit(0.036), restore(0.035), feature(0.031), sensitive(0.026), limitation(0.024), understand(0.022), reason(0.022), remove(0.021), inconvenience(0.020)
8	intend(0.036), contain(0.031), recipient(0.029), sender(0.027), copy(0.025), confidential(0.024), privilege(0.021), attachments(0.020), error(0.019), notify(0.018)
9	item(0.055), ihre(0.028), respond(0.023), policy(0.022), member(0.020), question(0.019), trademark(0.017), policies(0.017), register(0.016), learn(0.016)
10	usaa(0.092), apple(0.086), size(0.067), view(0.027), document(0.019), pending(0.018), payment(0.018), charge(0.017), correct(0.017), detail(0.015)
11	error(0.023), information(0.020), change(0.019), bill(0.017), western(0.014), slight(0.014), verification(0.013), personal(0.013), detect(0.013), accurately(0.013)

TABLE V: LDA - 25 Topics (continue...)

Topic	Words (Similarity Value)
12	hsbc(0.061), activity(0.021), bank(0.017), verify(0.016), josemonkey(0.015), make(0.015), attempt(0.015), restrictions(0.014), investigate(0.014), unsubscribe(0.014)
13	compte(0.099), informations(0.036), merci(0.031), cliquez(0.030), curit(0.026), jour(0.024), cette(0.023), notre(0.021), bonjour(0.020), veuillez(0.020)
14	assistance(0.047), add(0.037), corner(0.034), locate(0.031), monitor(0.031), response(0.029), agree(0.028), address(0.025), thec(0.024), page(0.023)
15	record(0.070), update(0.052), barclays(0.034), amazon(0.030), continue(0.025), come(0.022), normal(0.021), interrupt(0.020), online(0.020), future(0.019)
16	administrator(0.096), validate(0.096), mailbox(0.083), exceed(0.054), storage(0.053), able(0.0490, limit(0.038), google(0.033), mail(0.032), run(0.032)
17	union(0.022), credit(0.020), security(0.017), proceed(0.016), resolve(0.016), federal(0.016), alert(0.016), problem(0.014), suspend(0.013), center(0.012)
18	upgrade(0.028), servers(0.016), bank(0.015), spam(0.015), webmail(0.014), active(0.013), confirm(0.012), online(0.012), failure(0.011), detail(0.011)
19	access(0.022), limit(0.017), activity(0.017), security(0.016), remain(0.016), restore(0.015), measure(0.015), review(0.014), issue(0.014), understand(0.013)
20	enforcement(0.033), violation(0.032), commit(0.028), national(0.026), attempt(0.025), aware(0.021), amazon(0.021), relate(0.020), holder(0.019), theft(0.018)
21	refund(0.042), password(0.030), revenue(0.020), enter(0.017), sure(0.017), internal(0.016), tip(0.015), protect(0.014), info(0.014), securitytips(0.013)
22	webmail(0.024), update(0.017), password(0.017), warn(0.015), mail(0.015), confirm(0.012), lose(0.012), delete(0.011), owner(0.011), service(0.011)
23	america(0.045), bank(0.032), payment(0.017), online(0.015), amazon(0.015), senha(0.013), sign(0.012), bankofamerica(0.012), federal(0.012), information(0.012)
24	bank(0.024), business(0.016), form(0.015), money(0.015), customers(0.012), customer(0.012), online(0.011), free(0.010), program(0.010), start(0.009)

VI. Results & Analysis

A. LDA Results

Through the LDA modeling for generating labels of input data, we captured and compared 5, 10, and 25 topics. The result for the 5 and 25 topics are shown in Tables III, IV and V. The LDA model was able to give us topics but the topics that we obtained were not relevant and thus not cohesive. This could be observed by the very small values measured for each term in each topic in Tables III, IV, and V.

We could make some assumptions based on the topic, but the topics were very vague in 5 topic results, and we could not categorize them in Cialdini's principal category. With the 25 topics, the topics were not only vague but also broad. More specifically, some information could be retrieved from the topics but nothing discrete could be said. For example, in 25 topic examples, we could say that topic 2 is related to the threat but it does not align with Cialdini's principle.

B. Transformers' Results

For the transformer-based models, we calculated the precision, recall, and F1-score as our evaluation metrics for the classifiers. We trained each model for 50 epochs. After 50 epochs, there were not any substantial changes in the performance, as reflected in the evaluation metrics. The performance metrics are reported in Table VI.

As listed in Table VI, the RoBERTa model offers better performance than the other two transformers as indicated by the slightly better F1-scores reported. We also notice that the F1-scores obtained by RoBERTa are lower for authority (0.82) followed by Commitment (0.85), Social Proof (0.91), and then Reciprocity (0.93). The F1-score value for Scarcity is computed as 1.00, which might be an indication of overfitting and lack of enough data for this category that was remedied by oversampling.

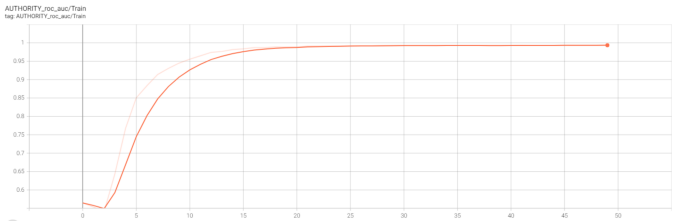
We also captured the training time for each of the models. The training time is shown in Table VII. The DistilBERT model was trained the fastest, while the BERT and RoBERTAa models have similar training times. Though the training times for RoBERTa and BERT were almost similar, we noticed that the RoBERTa model was the fastest to converge (i.e., stable).

C. ROC Curves

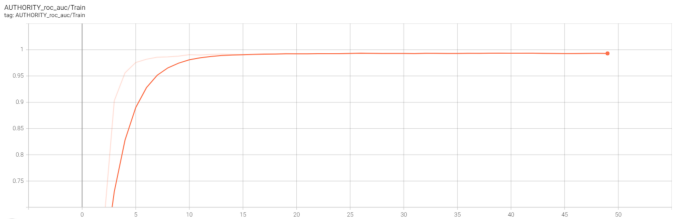
The plots illustrated in Figure 3 depict the ROC curves for each transformer model where the x-axis is the number

TABLE VI: The performance of the transformer-based models.

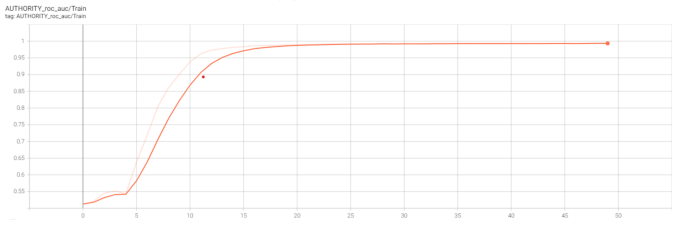
Persuasion Principle	BERT			RoBERTa			DistilBERT		
	Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
Reciprocity	1.00	0.79	0.88	0.88	1.00	**0.93**	0.97	0.81	0.88
Commitment	0.88	0.90	**0.89**	0.94	0.78	0.85	0.91	0.75	0.82
Scarcity	1.00	0.98	0.99	1.00	1.00	**1.00**	1.00	0.98	0.99
Authority	0.91	0.74	**0.82**	0.92	0.74	**0.82**	0.91	0.73	0.81
Social Proof	0.91	0.86	0.89	1.00	0.83	**0.91**	0.97	0.86	**0.91**
Average	0.94	0.85	0.89	0.94	0.87	0.9	0.95	0.82	0.88



(a) BERT



(b) RoBERTa



(c) DistellBERT

Fig. 3: The ROC curves captured for training the models for "authority" for 50 epochs.

TABLE VII: Training time of the models.

Model	Training Time
BERT	1 hour 41 minutes
RoBERTa	1 hour 40 minutes
DistelBERT	56 minutes

of epochs and the y-axis is the accuracy value. The plots show the ROC curves for training accuracy captured for "authority" during executing the training process several times (i.e., epochs). Each plot is annotated with two curves; one for training and the other one for validation accuracy captured during the training.

According to the plots in Figure 3, we observe that the RoBERTa model training time was much faster than the other models. The validation and training loss of the RoBERTa model at 12^{th} epoch was comparable to the training and validation loss of the DistilBERT model at 22^{nd} epoch. Also, we observe that the RoBERTa model can be trained with a lower number of epochs (15 epochs) to obtain a similar result at 50 epochs. As shown in the graph, the learning curve almost does not change after the 15^{th} epoch in the RoBERTa model.

VII. Threats to Validity

The experiments conducted and results reported in this paper are prone to a number of external and internal threats to validity that should be considered when replicating a similar experiment.

As indicated in Section V-A, the original dataset is an unlabeled one. We utilized BERT and a few samples to train a model to automatically label the dataset. In the experiments conducted in the work, We relied on 1) the accuracy of the implementation of BERT in performing classification and similarity matching, 2) the manual labeling performed on the initial set of samples as listed in Table I. Furthermore, the initial sample used for training the labeling model and the rest of the data are drawn from the same dataset. This issue may cause some issues related to generalization of the results and findings. Finally, it is important to note that the results obtained for LDA was based on the dataset used in this study. A different set of dataset or approach may result in different results.

VIII. Conclusion and Future Work

Phishing is the most popular social engineering attacks. In these attacks, the attacker crafts an email with the contents that are interesting enough to victims and thus lure them to comply with the requests stated in the email. It is important to learn about the mindset of attackers when crafting these emails so a better defending strategy can be deployed. The utilization of principles of persuasion [2] is the primary techniques in crafting such effective emails.

This paper investigates the use of persuasion principles in a phishing dataset. The objective is to develop a classification model that can detect the persuasion principle utilized in phishing emails with respect to the Cialdini's classification on persuasion principles [2] including Reciprocation, Consistency, Social Proof, Likeability, Authority, and Scarcity.

There exist a good number of machine learning-based classification models that can be adapted for the problem stated in this paper. Examples of such classification techniques span a wide range of conventional machine learning techniques (e.g., support vector machines) to more sophisticated techniques (e.g., CNN). In this paper, however, we investigate the possibility of developing an effective classifier using machine learning-based transformers (e.g., BERT), an emerging

language modeling techniques where the model utilizes an encoder-decoder architecture along a self-attention layer to perform the classification task.

The BERT transformer model and its optimized versions, such as RoBERTa and DistilBERT, are developed for language modeling. In this paper, we adapted these techniques to investigate whether it is possible to classify given emails/texts into the Cialdini's principles. The results obtained in our experiments show that these models can offer reasonable outcomes for this classification problem. The results obtained show that the RoBERTa model was slightly better than the basic BERT and also its optimized variation, DistilBERT. On the other hand, we observed that, due to optimization offered by DitilBERT, this type of transformer model is trained faster than the other two.

The primary issue in using these types of models is the availability of the labeled dataset. Our method of labeling sentences provided us a labeled dataset of about $3,924$ sentences/emails before balancing. This dataset was only enough to fine-tune the models. As part of future research work, we need to create a labeled dataset to train these models, which can certainly yield better results.

ACKNOWLEDGMENT

This work was supported in part by the U.S. National Science Foundation (NSF) under Grant 1723765.

REFERENCES

[1] L. F. Gutiérrez, F. Abri, M. Armstrong, A. S. Namin, and K. S. Jones, "Email embeddings for phishing detection," in *2020 IEEE International Conference on Big Data (Big Data)*, 2020, pp. 2087–2092.
[2] R. B. Cialdini, *Influence: The Psychology of Persuasion*. Harper Business, 2006.
[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.
[4] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," arXiv preprint arXiv:1907.11692, 2019.
[5] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," arXiv preprint arXiv:1910.01108, 2019.
[6] A. van der Heijden and L. Allodi, "Cognitive triaging of phishing attacks," in *28th USENIX Security Symposium (USENIX Security 19)*, 2019, pp. 1309–1326.
[7] M. Merchant, "Semantic similarity with bert," https://keras.io/examples/nlp/semantic_similarity_with_bert/, 2020.
[8] N. Akbar, "Analysing persuasion principles in phishing emails," Master's thesis, University of Twente, 2014.
[9] H. Nishikawa, T. Yamamoto, B. Harsham, Y. Wang, K. Uehara, C. Hori, A. Iwasaki, K. Kawauchi, and M. Nishigaki, "Analysis of malicious email detection using cialdini's principles," in *2020 15th Asia Joint Conference on Information Security (AsiaJCIS)*, 2020, pp. 137–142.
[10] "Enron email dataset," https://www.cs.cmu.edu/ enron/.
[11] T. Stojnic, D. Vatsalan, and N. A. G. Arachchilage, "Phishing email strategies: Understanding cybercriminals' strategies of crafting phishing emails," *Wiley Security and Privacy*, vol. 4, no. 5, 2021.
[12] S. Duman, K. Kalkan-Cakmakci, M. Egele, W. Robertson, and E. Kirda, "Emailprofiler: Spearphishing filtering with header and stylometric features of emails," in *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1, 2016, pp. 408–416.
[13] "Distilbert," https://huggingface.co/docs/transformers/model_doc/distilbert.
[14] J. Nazario, "Phishing dataset," https://monkey.org/ jose/phishing/phishing2.mbox.
[15] M. Merchant, "Semantic similarity with bert," https://keras.io/examples/nlp/semantic_similarity_with_bert/.
[16] K. S. Jones, M. E. Armstrong, M. K. Tornblad, and A. S. Namin, "How social engineers use persuasion principles during vishing attacks," *Information and Computer Security*, vol. 29, no. 2, pp. 314 – 331, 2020.
[17] "Genism topic model for humans," https://radimrehurek.com/gensim/.
[18] "SpaCy API documentation," https://spacy.io/api/doc, 2022.