Persuasion Detection and Continuous Bag of Words for Detecting Phishing Emails

Robert Pauls
210018702
MSci Computer Science
robert.pauls@city.ac.uk

1 Problem statement and motivation

With the shift to a digitised society has come our reliance on online platforms for contact across the internet. Platforms such as email provide a proxy for the delivery of messages of malicious intent aiming to hoax users into giving up sensitive information. With attacks targeting industries to individuals, the dynamism of these attacks is displayed in the constant emergence of new linguistic strategies that exploit human emotions with the objective of extracting sensitive account details.

In a comprehensive report by Smith (2024), email is found to be the most popular method for conducting phishing attacks across the internet with an estimated 91% of cyber attacks being initiated with a phishing email. Google Mail is the most commonly sought platform due to its ability to create multiple accounts very quickly and for free, along with their "read receipts" function. An estimated 3.4 billion emails are thought to be delivered each day posing as a trusted sender.

Industries as whole have also been attacked by these emails with the objective of extracting large sets of private data. Ending 2022, a total of 4,744,699 phishing attacks were observed with a notable 27.7% of the attacks directed at financial institutions highlighting the targeted nature of these cyber threats.

By detecting and dissecting the various linguistic tactics of phishing emails, we gain the power to educate users of the internet on how criminals take advantage of their innocence and unawareness to the power of social engineering tactics.

2 Research hypothesis

Can we correctly classify phishing emails using features extracted with Continuous Bag of Words? Phishing emails are composed in a persuasive yet seemingly innocent way that fools unknowing users, making use of various terminology in large sums. They come in similar structures each depending on the objective.

The proposed method aims to harness the vast range of persuasive techniques, as described in Dimitrov et al. (2021), to effectively identify and analyse persuasiveness within phishing emails.

CBOW strives through its ability to generate word embeddings based on the ordering of context, that are used to capture semantic relationships between words. Contextual understanding is a crucial part in being able to discern subtle linguistic cues.

3 Related work and background

Detection of phishing emails has been widely researched in the past few years, generating many unique ideas to solve the problem. Puppeteer, as proposed by Cho et al. (2021), presents a hybrid system integrating finite state transducers to orchestrate the pushing of agendas, and a neural dialogue system generating responses to the prompts. Similarly, Shahriar et al. (2021) delve into the effects of deception on a domain-independent basis, leveraging deep learning architectures such as BERT, ensemble models and character level-CNNs. In the realm of active defence, Dalton et al. (2020) describes Panacea, an active defense system for social engineering attacks. The system utilises NLP technologies to process modern message formats for accommodating innovative approaches such as analysis of messages, dialogue generation and knowledge representation.Additionally, Chatterjee and Basu (2021) incorporate BERT models, including DistilBERT, within their neural network framework, preferring transformers over RNNs for their self-attention mechanism ((Chatterjee and Basu, 2021)). This mechanism enables the decoder to iteratively select pertinent information during decoding, enhancing the model's efficacy. Todorovic et al. (2023) hypothesise various approaches for classifying email topics. Abu-Nimeh et al. (2007) compare supervised learning algorithms like logistic regression, random forest, and SVM, demonstrating their effectiveness in detecting phishing emails using bag-of-words models with TF-IDF weights (Abu-Nimeh et al., 2007). Diale et al. (2019) implement a set of non-linear machine learning algorithms, including Random Forest and decision tree algorithms, combined with vector size reduction to eliminate excessive numbers of features. Dimensionality reduction is utilised to capture word ordering and basic semantic meaning from text messages. Results show an overall detection accuracy of 97% over the Enron dataset. Hovold (2006) further explores the use of a Naive Bayes classifier as a basis for personalised spam filters, augmented by attribute selection, n-grams and cost-sensitive classification. Castillo et al. (2020) adopt a machine learning approach, employing a set of neural network architectures and evaluating the result of applying pre-trained word embedding representations to these architectures for accurately classifying malicious emails. A Continuous Bag of Words model based on Gensim-word2vec is used for obtaining counts of words, with a window of 10 words for analysing neighbourhood of texts and various size vectors created to test each neural network architecture. Dimitrov et al. (2021) discusses some of the persuasive techniques used in attempts to successfully perform phishing. The most relevant of these to this paper's theme include loaded language, labelling, exaggeration and appeal to fear or prejudices.

3.1 Accomplishments

Below is a list of the proposed tasks for this paper. Some further tasks have been added from the proposal to help ensure the most accurate results can be gained from the primary model.

- **Preprocessing**: Preprocess the dataset for better compatibility with task - completed
- **Tokenisation**: Perform tokenisation of cleaned data and generate corpus - completed
- **Baseline Models**: Build, train and tune baseline models and examine performance - completed
- **Generate Word Embeddings**: Build and train CBOW model to generate word embeddings - completed
- **Develop Persuasion Detection Model**: Build, train and test model for persuasion detection and examine performance in comparison to baseline models - failed; while simple classification of phishing emails was performed, it did not meet the objective of utilising persuasive language to perform the classification.
- **Model Tuning**: Perform hyperparameter tuning for primary models and examine performance in comparison to baseline models - completed
- **Critical Analysis of Results**: Perform in-depth error analysis to figure out what kinds of examples the described approach struggles with - completed

4 Approach and Methodology

The approach used for this paper was to apply various preprocessing techniques such as removal of stopwords and other unnecessary text, and generate word embeddings from the final cleaned dataset. Through this process, intricate semantic relationships between words in the text are captured generating the word embeddings that serve as the foundation for the training of the persuasion detection model and analyse the results of the model. Following training, an in-depth analysis is performed using the obtained metrics and evaluation of the final results.

The described approach leverages the power of word embeddings for encapsulating semantic relationships between words consequently enhancing the textual content. Unlike word embeddings, the baseline models don't possess the ability to capture intricate nuances of language and context limiting their effectiveness in discerning persuasive techniques. The reliance on word embeddings generated solely from the CBOW model may result in a loss of information related to the broader semantic context, particularly in instances where complex syntactic structures or long-range dependencies are crucial for understanding persuasive language.

While a working implementation was completed, it did not meet the initially proposed objectives. The generated word embeddings were used to perform classification of phishing emails, however were not utilised to detect the identified persuasive language of previous works used in the phishing emails.

The libraries listed below were used to implement the described approach

- os for file handling

- re for creating and utilising regular expressions for preprocessing
- Pandas for understanding and performing preprocessing on data
- nltk for tokenisation and removal of stop words and non-words
- Sklearn for vectorisation of data, splitting data for training and testing, developing baseline and persuasion detection models, and generating model statistics
- PyTorch for creating and training Word2Vec CBOW deep learning model

The basis of the CBOW model is taken from `https://colab.research.google.com/github/pmadhyastha/INM434/blob/main/distributional_models.ipynb`. Hyperparameter tuning is performed by hand, and batch learning implemented to accelerate the training process.

The baseline models were developed by hand, using documentation provided by Sklearn. These can be found in the Baseline Models section of the provided Phishing Detection.ipynb notebook.

The primary models were developed by hand, using documentation provided by Sklearn. These can be found in the Primary Models section of the provided Phishing Detection.ipynb notebook.

5 Dataset

5.1 Introduction

The dataset consists of over 18000 safe and phishing emails generated by Enron Corporation employees (Chakraborty, 2023). A CSV file of size 52.03MB is provided, 'Phishing_Email.csv', consisting of 18650 rows with 3 features:

- an unnamed column, of type identifying the row number
- 'Email Text', of type 'object', representing the email body
- 'Email Type', of type 'object', identifying whether an email is legitimate or attempted phishing. 61% (11322) of the emails constitute safe emails while the remaining 7328 emails constitute phishing emails

The email text consists of a broad range of characters, various word formations such as abbreviations, acronyms and clippings, URLs and email headers. Some of the text is written in a poor grammatical manner and, in some cases, includes sequences of characters that have no obvious meaning. Some samples of the non-preprocessed data can be seen below.

- re : 6 . 1100 , disc : uniformitarianism , re : 1086 ; sex / lang dick hudson 's observations on us use of 's on ' but not 'd aughter ' as a vocative are very thought-provoking , but i am not sure that it is fair to attribute this to " sons " being " treated like senior relatives " . for one thing , we do n't normally use ' brother ' in this way any more than we do 'd aughter ' , and it is hard to imagine a natural class comprising senior relatives and 's on ' but excluding ' brother ' . for another , there seem to me to be differences here .
- software at incredibly low prices (86 % lower) . drapery seventeen term represent any sing . feet wild break able build . tail , send subtract represent . job cow student inch gave . let still warm , family draw , land book . glass plan include . sentence is , hat silent nothing . order , wild famous long their . inch such , saw , person , save . face , especially sentence science . certain , cry does . two depend yes , written carry .
- earnings calger $ 42 pastoria $ 7 q 2 & $ 10 q 3 fountain valley $ 2 7 eas $ 3 . 5 delta $ 7 m cdwr option $ 80 m to date
- The academic discipline of Software Engineering was launched at a conference sponsored by NATO, at Garmisch, Germany, in October, 1968. Intriguingly, the term Software Engineering was chosen to be deliberately provocative – why can't software be developed with the same rigor used by other engineering disciplines?The proceedings of this conference are now available online, at: http://www.cs.ncl.ac.uk/old/people/brian.randell/home.formal/NATO/index.htmlAlso, don't miss the pictures of attendees, including many significant contributors to the field of Software Engineering: http://www.cs.ncl.ac.uk/old/people/brian.randell/home.formal/NATO/N1968/index.html-Jim

Some problems presented by the dataset include:

- **Poorly written texts**: A majority of the text is written in a poor grammatical manner, requiring the need for careful preprocessing and presenting a possible challenge for classification.

- **Random character sequences**: A lot of random character sequences are discovered which introduce noise in the dataset that likely harm the vectorisation process.

- **Imbalanced class distribution**: The number of safe emails slightly outweighs the number of phishing emails, most likely resulting in biased predictions where the model will fail to accurately classify phishing emails. This is discussed further in the following section.

These problems emphasise the need for extensive examination of the dataset and application of various preprocessing techniques.

5.2 Preprocessing

The final preprocessing techniques used for the data include:

- **Lowercasing**: All text is converted to lowercase, a typical step in all NLP tasks to ensure consistency during model development.

- **Removing line breaks**: Line breaks carry no semantic meaning and harm the classification process.

- **Removing stop words**: Stop words provide no meaning to the text; removing these assists in noise reduction, helping the model focus on more meaningful text.

- **Removing non-word and digit characters**: Special characters introduce noise in the data and provide no information value for the task at hand.

- **Removing non-words**: WordNet is used to eliminate any non-words; these harm the CBOW model's ability to generate meaningful context and are therefore removed.

Limiting the preprocessing helps to ensure the models can consider all context, while preventing noise from disturbing the final results of the implemented models.

6 Baselines

Three models are used as a baseline; Decision Tree, Support Vector Machine and Naive Bayes. Implementing multiple baseline models ensures various factors such as efficiency, complexity or robustness can be assessed effectively when developing further models.

Decision trees offer a straightforward way of understanding relationships between text data. They can capture complex relationships between features, especially non-linear relationships such as in the case of this task.

SVMs are effective in handling high-dimensional feature vectors, a challenge represented by the dataset used in this paper. Being less prone to overfitting ensures it can maximize the margin between various classes allowing for better generalisation performance.

Naive Bayes, particularly Multinomial NB, classifiers are highly efficient with quick training time, suitable for large datasets. Its robustness to irrelevant features means even if some features aren't relevant, such as those that may not be removed in preprocessing, Naive Bayes is still able to perform well.

Model	**Precision**	**Recall**	**F1-Score**	**Accuracy**
DT	71%	71%	71%	73%
SVM	97%	97%	97%	97%
MNB	97%	97%	97%	97%

Figure 1: Baseline models performances

Training the baselines is performed using 80% of the dataset while the remaining 20% is used for testing.

7 Results and Error Analysis

Figure 2 reports on the performance of applying the embeddings generated for the corpus to two non-linear models. Although the word embeddings were generated using 40% of the total corpus - this helped accelerate the CBOW training process - the results show the embeddings were able to effectively capture the semantic information.

Training the non-linear models is performed by loading and aggregating the retrieved embeddings to all the emails, and feeding these to the models

Model	**Precision**	**Recall**	**F1-Score**	**Accuracy**
RF	93%	92%	92%	93%
MLP	96%	95%	95%	96%

Figure 2: Primary models performances

Both the models perform better than the decision tree in all categories, however not as much when compared to the SVM and Naive Bayes classifiers. This is most likely attributed to the limited corpus usage which restricts the ability of the models to capture full semantic richness of the data. The results presented strongly validate the efficacy of employing word embeddings in conjunction with machine learning models for classifying phishing emails.

Manual error analysis shows the baseline models seemed to fail with inputs that contained technical or jargon-heavy language, or URLs. Some of these texts are seen below:

- oddly enough discusing www geocaching com opinons range yeaboy itsagainstthelaw mostly though either folks boned subject cool sheeple bleating big daddy protect sstill cant get head around fact many people much willing sheeple things like patriot act surrounding shackle racking http opentopic groundspeak com opentopic tpc f http xent com mailman listinfo fork

- quality life decisions firings dudley bus kelly dull dynamism loitering banal dissension estop capaciousness weierstrass myofibril imbecile stygian appreciate ta hydraulic decomposable hindrance irishman hidalgo affection arrhenius mathematician houseboat disillusioned harmony analogies audited cheat segovia lanky dimness knaves procter justifies modally astigmat framed hellenic bookish laszlo assembles alumnus depositions inanimately coot malts alp nativity inflicting lionesses ensured improve dialogue hurley hamburger databases archeologist

- url http jeremy zawodny com blog archives html date noted inluminent weblog seem many bloggers osxcon least blogging rather disappointing strong oscon emerging technologies conference damn wish foresight

The primary models seemed to perform no better in these aspects. This is most likely due to the properties discussed creating noise that harms the classification process.

8 Lessons learned and conclusions

The results show part of the proposed methodology applied was effective, however the whole implementation requires much more work to be fully complete and provide more accurate results.

Performing this project has given me insight into developing a text classification pipeline. The importance of tokenisation and vectorisation of data when developing text classification models is crucial to ensuring the most effective results are produced. The use of baseline models has helped me understand how baselines can be used to assess the performance of complex models. Large datasets introduce extremely long run times; running the cleaned dataset of 18101 rows would've taken more than a day to be performed without feeding the model with batches.

Some of the most useful techniques used include:

- Regular expressions for detecting features of text data - significantly reduced manual inspection

- Batch learning for training CBOW model; reduced time for processing significantly compared to directly feeding whole corpus

Some issues encountered during the project included:

- When attempting to generate training and testing sets using the vectorised corpus, an inconsistent numbers of samples was found. In further investigation, this was due to incorrect generation of the corpus.

- The dataset seems to have some data that wasn't formatted correctly. When performing error analysis for the baseline models, some of the data had the incorrect actual label associated to it. In further work, this would need special measures to be handled.

Working with text-based datasets requires a lot of attention during preprocessing to ensure the most relevant data can be extracted and to help reduce the noise that affects the classification process.

The originally proposed goals were not achieved; although attempts were made to use the obtained embeddings to classify persuasive language in phishing emails, I ended up only with classification of phishing emails. Further work would be required to achieve the original goal.

References

Saeed Abu-Nimeh, Dario Nappa, Xinlei Wang, and Suku Nair. 2007. A comparison of machine learning techniques for phishing detection. In *Proceedings of the Anti-Phishing Working Groups 2nd Annual ECrime Researchers Summit*, eCrime '07, page 60–69, New York, NY, USA. Association for Computing Machinery.

Esteban Castillo, Sreekar Dhaduvai, Peng Liu, Kartik-Singh Thakur, Adam Dalton, and Tomek Strzalkowski. 2020. Email threat detection using distinct neural network approaches. In *Proceedings for the First International Workshop on Social Threats in Online Conversations: Understanding and Management*, pages 48–55, Marseille, France. European Language Resources Association.

Subhadeep Chakraborty. 2023. Phishing email detection.

Anik Chatterjee and Sagnik Basu. 2021. How vulnerable are you? a novel computational psycholinguistic analysis for phishing influence detection. pages 499–507.

Hyundong Cho, Genevieve Bartlett, and Marjorie Freedman. 2021. Agenda pushing in email to thwart phishing. In *Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering (DialDoc 2021)*, pages 113–118, Online. Association for Computational Linguistics.

Adam Dalton, Ehsan Aghaei, Ehab Al-Shaer, Archna Bhatia, Esteban Castillo, Zhuo Cheng, Sreekar Dhaduvai, Qi Duan, Bryanna Hebenstreit, Md Mazharul Islam, Younes Karimi, Amir Masoumzadeh, Brodie Mather, Sashank Santhanam, Samira Shaikh, Alan Zemel, Tomek Strzalkowski, and Bonnie J. Dorr. 2020. Active defense against social engineering: The case for human language technology. pages 1–8.

Melvin Diale, Turgay Celik, and Christiaan Van Der Walt. 2019. Unsupervised feature learning for spam email filtering. *Computers & Electrical Engineering*, 74:89–104.

Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. 2021. SemEval-2021 task 6: Detection of persuasion techniques in texts and images. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 70–98, Online. Association for Computational Linguistics.

Johan Hovold. 2006. Naive bayes spam filtering using word-position-based attributes and length-sensitive classification thresholds. page 78–87.

Sadat Shahriar, Arjun Mukherjee, and Omprakash Gnawali. 2021. A domain-independent holistic approach to deception detection. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1308–1317, Held Online. INCOMA Ltd.

Gary Smith. 2024. Top phishing statistics for 2024: Latest figures and trends.

Branislava Sandrih Todorovic, Katarina Josipovic, and Jurij Kodre. 2023. Three approaches to client email topic classification. page 1015–1022.