# Manipulative Language Detection in LLM-Crafted Phishing Attacks

Karl-Johan Westhoff
email: kjwesthoff@berkeley.edu
Neha Dhage
email: neha_dhage@ischool.berkeley.edu

UC Berkeley School of Information
MIDS Course 266 Summer 2025 Section 2 (Natalie Ahn)

## 1 Introduction

The human factor remains central in cyber attacks. The 2024 Verizon DBIR report [1] notes that 68% of breaches involve the human element, with phishing as a key contributor. With LLM tools, bad actors can now craft highly convincing phishing messages that evade traditional detection. This project investigates whether NLP models can detect manipulative language—specifically, text designed to influence actions not in the reader's best interest.

Machine learning (ML) models like Naive Bayes and basic neural networks are widely used to filter email traffic for spam (which is an abundant problem). However, they are often limited to detecting specific words or obvious patterns. Newer approaches combine lightweight ML filtering with resource-heavy NLP methods for cases that are not clearly categorized by simpler filtering. Since phishing often exploits human psychology through language, this study focuses on detecting manipulative language and whether such detection may improve defenses against phishing. Although the focus is on cybersecurity, manipulative language also appears in areas such as coercive or abusive communication, highlighting its broader relevance. Our approach first models manipulation using the "Mental Manip" dataset, then explores its potential for phishing detection.

## 2 Literature

Salloum et al. [2] provide an overview of current ML and NLP methods used for phishing detection, which forms the foundational context for this project. Suhaima et al. [3] trained models like BERT on spam data, whereas our focus will be on specifically detecting manipulative language. Wang et al. [4] created a data set aiming at dialogue manipulation, which will serve as our primary training set. Al-Subaiey et al. have compiled a large corpus of emails in [5] from various datasets, under phishing-specific email body texts; this will be used for attempts to detect phishing texts.

# 3 Datasets

Labeled data sets focused on manipulation are rare. Most of the research has come from psychology, which provides insight into the techniques used for manipulation rather than bulk data suitable for AI model training. Most existing data sets suitable for NLP applications are concerned with hate speech and abusive language, which has been an important topic in relation to social media.

# 4 The MentalManip Dataset

Wang et al. [4] introduced the "MentalManip" dataset, published on hugging face [6]. The data set is based on fictional dialogues from "The Cornell Movie Dialogs Corpus" [7] from which suitable manipulative dialogues were selected using BERT and GPT-4 models, from these, 4000 dialogues were manually selected to form the data set. The data is labeled with a detailed manipulation taxonomy in three dimensions; see Figure 1, adding applied technique and psychological vulnerability mechanism to the binary presence of whether the dialogue contains manipulation or not.

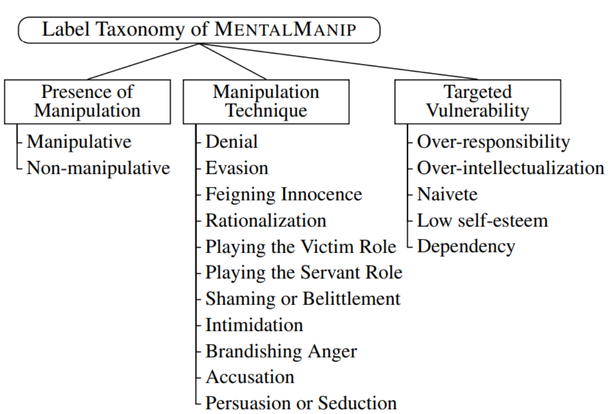



**Figure 1:** *Taxonomy labels in the data set*

The data set was manually labeled using a multi-phase human annotation process, adapting the taxonomy (Figure 1) to the dialogue context three times by different people annotating. This gave two versions of the data set, one where the majority two out of three constitutes the result ("$MentalManip_{maj}$") and one where all three annotators have consensus and reach the same results ("$MentalManip_{con}$"). The $MentalManip_{maj}$ data set is larger (4000 rows) and more suitable for training a model capturing more instances of manipulation, the $MentalManip_{con}$ data set is smaller (2920 rows) and more precise and better suited for fine tuning. For this project we used the $MentalManip_{maj}$ data.

## 4.1 Data Exploration

In some cases these data fields are not complete in the data set requiring some degree of feature manipulation, This is addressed in section **??** below.

## 4.2 Dialogues

The 4000 dialogues in the data set are between two persons exchanging sentences. By far the majority of dialogues consist of two exchanges, one by each person (there are only three cases with three exchanges). Word count statistics are shown in Figure 2, most dialogues consist of up to 50 words per person, and the number of words uttered by each person is fairly balanced, with person 2 saying slightly more words than person 1 in the up to 50 word majority case. Figure 3 shows the distribution of token counts for the dialogues in the data set, tokenized using BERT-base as reference. Only a minor number of dialogues exceed the BERT-base embedding size of 512 tokens.

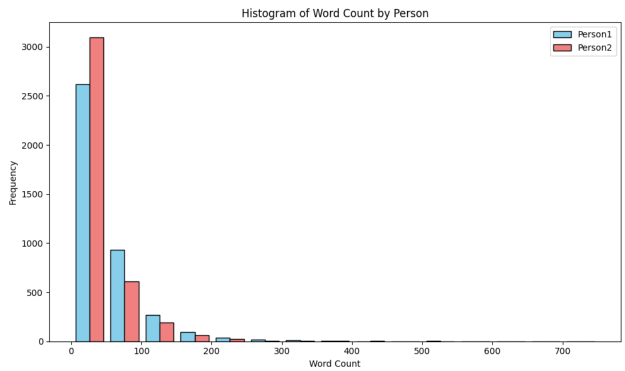



**Figure 2:** *Word count statistics for the dialogues in the MentalManip$_{maj}$ data set, words uttered by each person*

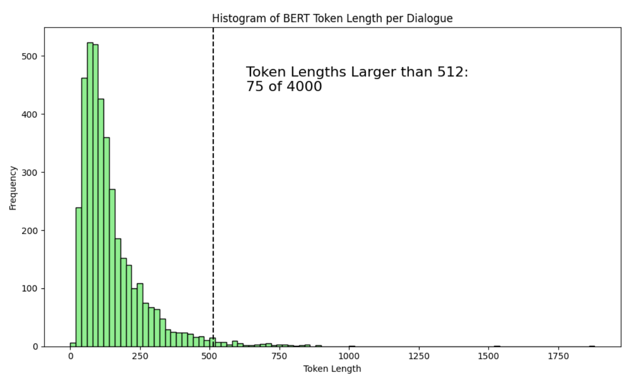



**Figure 3:** *Statistics for the dialogue in the MentalManip$_{maj}$ data set, tokenized using BERT-base*

## 4.3 Labels

**Manipulation Label** The data set is not split equally between manipulation and non-manipulation, Figure 4 shows the distribution with 2.4 times more manipulation rows than non-manipulation (discussed in section **??**).

**'Technique' and 'Vulnerability' Labels** Some of the labels are missing for some of

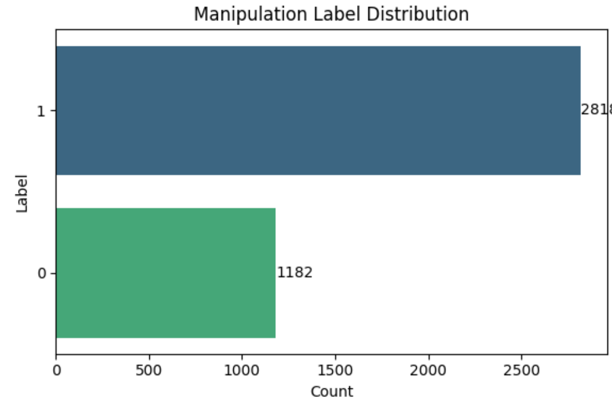



**Figure 4:** *Ratio of manipulation to non-manipulation in the MentalManip$_{maj}$ Dataset*

the rows with manipulation present[1], Figure 5 shows a total of 664[2] missing labels for 'technique', we regard the technique labels as most relevant for phishing, especially the 'Persuasion or Seduction' label.

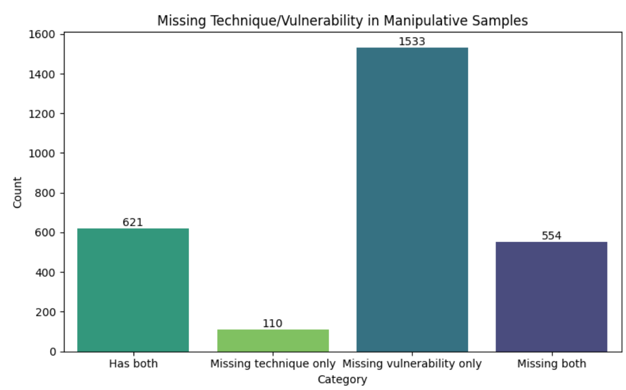



**Figure 5:** *Incomplete labeling of the MentalManip Dataset*

Further data exploration can be found in appendix A

[1] The labels should not be populated for non-manipulation rows

[2] 110 missing technique and 554 also missing vulnerability

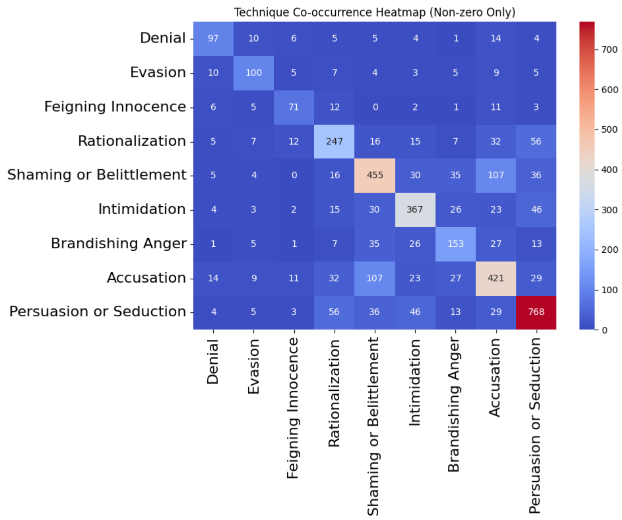



**Figure 6:** *Distribution and Co-ocurrence of technique labels*

# 5 Baselines

With the relatively short embeddings (see Figure 3), the more basic versions of BERT have sufficient capacity to handle the data. The MentalManip article [4] also uses some decoder only models by 'zero' and 'few-shot' prompting the model with random example from the data set. This seems to perform better for overall binary classification, but only a little, and the LLM's have a tendency to pick up on toxicity and hate-speech and identify these as manipulation. Considering the label inconsistencies for 'technique' and 'vulnerability' in the data set, we will focus on the binary classification of manipulation for choosing a baseline model for further experimentation.

## 5.1 Binary with BERT and Buddies

Models looking at the 'manipulative' labels are trained on the $MentalManip_{maj}$ data set. The models were run with similar parameters, and the Accuracy at epoch before significant over fitting[3] recorded. Following models were investigated:

- BERT-base [8]
- RoBERTa [9]
- DistilBERT [10]
- ModernBERT [11]
- DeBERTaV3 [12]

Furthermore some "emotionally wiser" BERT derivatives exist which are pre-trained for emotion detection:

- BERTweet [13]
- EmotionBERT [14]

## 5.2 Baseline Results and Discussion

Results with losses and accuracy are shown in Table 1. The models were run until significant over-fitting occurred. In general the models over-fit after a few epochs which is to be expected with a model that is extended from pre-trained. The Accuracy results are around 0.70-0.72 with little variation. Models can be found in Appendix B

**ModernBERT** The model does not perform better, this was expected as the embedding lengths are short (see Figure 3) and not leveraging the benefits of ModernBERTs larger capacity.

**Emotionally intelligent BERT** BERTweet performs on par with BERT-base, the model is primarily trained for "Part-of-speech tagging", "Named-entity recognition" and "text classification" [13] herunder including emjojis etc. i.e. The model is not per-se expected to be better at manipulation detection, but we thought to give it a

[3] Significant over fitting defined as: training loss / evaluation loss < 0.6

try. The EmotionBERT model was created for multi label classification of well known emotional phrases for media monitoring [14].

**Advanced BERT** We also tried some more advanced BERT derivatives (DestilBERT and deBERTa_v3_small), however these models did not perform better than RoBERTa, and they required more compute resources to train. DeBERTa uses more advanced training loss, pre-training more advanced encoding etc.[12], however only the smallest version of deBERTa was possible to train with the hardware available. These models are optimized to deliver faster inference, but the extra cost in training resources make them less feasible for this project.

**RoBERTa** The best performing model reported in Table 1 was RoBERTa, which seems to perform slightly better than BERT-base, however with multiple tries the performance was not consistent, sometimes BERT performed better, however RoBERTa seemed more stable giving consistent results above 0.72 and performing more Epochs before over-fitting.

# 6 Fine Tuning

Based on the results in section 5.2, we decided to fine tune the RoBERTa model with the $MentalManip_{maj}$ data set to see if we can get better results for the manipulation detection task. The MentalManip article[4] achieved an accuracy of 0.78 which in itself is not impressive.

| Model | Epoch | Loss T/V | Acc Epoch | Acc Final |
|---|---|---|---|---|
| BERT | 2 | 0.97 | 0.726 | 0.70 |
| roBERTa | 4 | 1.03 | 0.728 | 0.73 |
| deBERTa_v3 | 3 | 0.68 | 0.704 | 0.72 |
| DistilBERT | 2 | 0.75 | 0.709 | 0.72 |
| ModernBERT | 2 | 0.81 | 0.718 | 0.72 |
| BERTweet | 2 | 0.97 | 0.705 | 0.70 |
| EmotionBERT | 2 | 1.04 | 0.704 | 0.70 |

**Table 1:** *Base Model performance comparison across different transformer architectures for binary inference on the "manipulative" column: Epochs before significant over-fitting, Training loss / Validation loss (to measure overfitting) and accuracy at epoch and final classification*

## 6.1 LoRA/PEFT

## 6.2 Hyper parameters

# 7 Experiments

Persuasion is expected to be the main technique in phishing, where the aim is to get people to take some action on behalf of the attacker. Therefore we will focus on the 'persuasion' label in our experiments.

## 7.1 Persuasion

**Feature Engineering** The 'technique' column in the data set was chosen as binary label, with 'persuasion' present in the text being the positive class (some rows have multiple technique labels). Links to notebook in Appendix C.2. The label distribution is 4.2 to 1 for the 'non-persuasion' to 'persuasion' labels, i.e. near the opposite of the data sets 'manipulative' vs 'not' when including all techniques and vulnerabilities.

**Results and Discussion** Classification Report for the 'Persuasion' label, Base Case un-weighted is shown in Table 2. At a glance, the model shows decent overall accuracy (0.75), However as we are interested in identifying persuasion the important metrics are recall and precision for the 'Manipulative' class, here the results are not so impressive, out of all actual manipulative instances, only 43% were correctly predicted as manipulative (confusion matrix and results are shown in appendix C.2. A possible cause for the models difficulty identifying persuasion is the skewed label distribution.

| Class | Prec. | Rec. | F1 | Sup. |
|---|---|---|---|---|
| Non-manip. | 0.84 | 0.83 | 0.84 | 631 |
| Manipulative | 0.40 | 0.43 | 0.41 | 169 |
| **Accuracy** | | | 0.75 | 800 |
| **Macro avg** | 0.62 | 0.63 | 0.63 | 800 |
| **Weighted avg** | 0.75 | 0.75 | 0.75 | 800 |

**Table 2:** *Classification Report for the 'Persuasion' label, Base Case un-weighted*

## 7.2 Weighted loss function

«««< HEAD To remedy the skewed label distribution, a suggested approach is adding weights to the loss function, suppressing the majority and boosting the minority class. Alternatives are over- and under-sampling. Under-sampling was disregarded due to the limited dataset size of 4,000 rows, as it would further reduce the available training data and risk losing information. Oversampling was considered but disregarded, duplicating minority samples can lead the model to remember specific sentences. Zhang et al. [15] suggests that weighting the loss function is more effective than sampling when fine-tuning BERT on imbalanced data.
======= To remedy the skewed label distribution, a suggested approach is adding weights to the loss function, suppressing the majority and boosting the minority class. Alternatives are over- and under-sampling. Under-sampling was disregarded due to the limited dataset size of 4,000 rows, as it would further reduce the available training data and risk losing information. Oversampling was considered but ultimately avoided to prevent overfitting, particularly since duplicating minority samples can lead the model to remember specific sentences. Zhang et al. [15] suggests that weighting the loss function is more effective than sampling when fine-tuning BERT on imbalanced data.
»»»> origin/overleaf-2025-07-25-0407

**Results and Discussion** Weights are calculated based on the label distribution and applied to the cross-entropy loss function during training[4]. We tried a number of schemes to determine weights:

1. Weights inversely proportional to the class distribution, see Figure 7
2. Weights inversely proportional to the class distribution but max weight capped 4.0 and 3.0 respectively
3. Weights normalized to add up to 1

«««< HEAD The un-capped un-normalized option 1 version performed best, it adds the most weight to the minority class and gives a recall of 0.51 for the 'Manipulative' class (a slight improvement compared to the un-weighted in Table 2), results option 1 are shown in Figure 8. Results for option 2 and 3 gives slightly

[4] A custom derivative of the Trainer class was implemented, see Appendix C.3

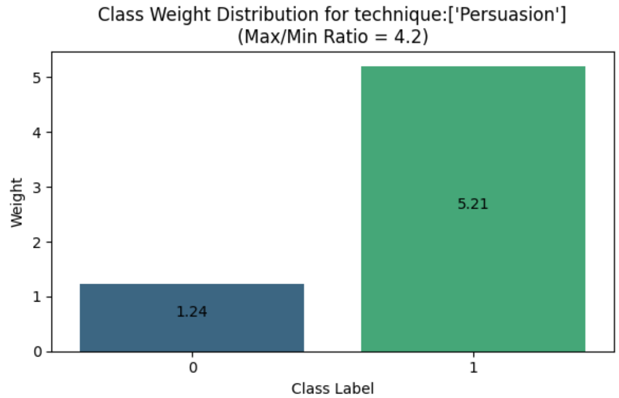
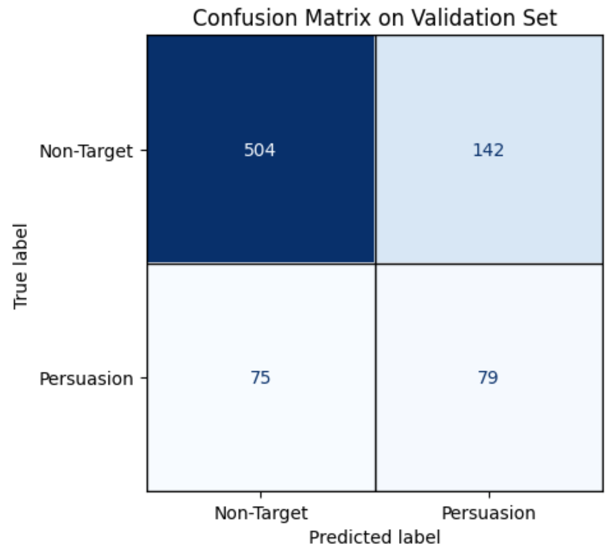



**Figure 7:** *Distribution of weights for the Persuasion label, no modifications to the weight distribution*

lower performing recalls in proportion to the weights added to the minority label and better overall Accuracy results (favoring the non manipulative class). Results are shown in Appendix C.3. A general observation during training was that training loss dropped as expected but the validation loss was rising although F1 score was still improving. The models used early stopping based on F1 score, even though the training loss vs test loss indicated over fitting. ======= The un-capped un-normalized option 1 version performed best, it adds the most weight to the minority class and gives a recall of 0.51 for the 'Manipulative' class (a slight improvement compared to the un-weighted in Table 2), results option 1 are shown in Figure 8. Results for option 2 and 3 gives slightly lower performing recalls in proportion to the weights added to the minority label and better overall Accuracy results (favoring the non manipulative class). Results are shown in Appendix C.3
»»»> origin/overleaf-2025-07-25-0407




Accuracy Score: 0.73

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Non-Target | 0.87 | 0.78 | 0.82 | 646.0 |
| Persuasion | 0.36 | 0.51 | 0.42 | 154.0 |
| macro avg | 0.61 | 0.65 | 0.62 | 800.0 |
| weighted avg | 0.77 | 0.73 | 0.75 | 800.0 |

**Figure 8:** *Results for un-capped and un-normalized weighted cross-entropy loss function*

# 8 Evaluation

Our main interest is to investigate if the model can extend existing phishing detection systems by detecting manipulating language in the emails. We will to look at false negative results from previous models, to see if the detection of manipulative text captures emails that were previously missed.

# 9 Conclusion

The baseline investigations showed that applying the *MentalManip* dataset to models optimized for emotional language detection does not automatically improve accuracy. Mental manipulation is less explored in literature than hate speech and abusive language, which are key concerns in social media, where sentiment analysis is both established and evolving in NLP.

# References


[1] Verizon Business. *2024 Data Breach Investigations Report*. Tech. rep. Accessed: 2025-05-21. Verizon, 2024. URL: https://www.verizon.com/business/resources/reports/2024-dbir-data-breach-investigations-report.pdf.

[2] Said Salloum et al. "Phishing Email Detection Using Natural Language Processing Techniques: A Literature Survey". In: *Procedia Computer Science* 189 (2021). AI in Computational Linguistics, pp. 19–28. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2021.05.077. URL: https://www.sciencedirect.com/science/article/pii/S1877050921011741.

[3] Suhaima Jamal, Hayden Wimmer, and Iqbal Sarker. *An Improved Transformer-based Model for Detecting Phishing, Spam, and Ham: A Large Language Model Approach*. Nov. 2023. DOI: 10.21203/rs.3.rs-3608294/v1.

[4] [Yuxin Wang, Ivory Yang ASD Saeed Hassanpour, and Soroush Vosoughi]. "MentalManip: A Dataset For Fine-grained Analysis of Mental Manipulation in Conversations". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2024, pp. 3747–3764. URL: https://aclanthology.org/2024.acl-long.206.

[5] Abdulla Al-Subaiey et al. *Novel Interpretable and Robust Web-based AI Platform for Phishing Email Detection*. 2024. arXiv: 2405.11619 [cs.LG]. URL: https://arxiv.org/abs/2405.11619.

[6] Yuxin Wang Ivory Yang Saeed Hassanpour Soroush Vosoughi. "MentalManip: A Dataset For Fine-grained Analysis of Mental Manipulation in Conversations". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2024, pp. 3747–3764. URL: https://huggingface.co/datasets/audreyeleven/MentalManip.

[7] Cristian Danescu-Niculescu-Mizil and Lillian Lee. "Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs." In: *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*. 2011.

[8] Google AI. *BERT-base*. https://huggingface.co/google-bert/bert-base-uncased. Accessed: 2025-07-21. 2019.

[9] Facebook AI. *RoBERTa*. https://huggingface.co/FacebookAI/roberta-base. Accessed: 2025-07-21. 2019.

[10] Hugging Face. *DistilBERT*. https://huggingface.co/distilbert-base-uncased. Accessed: 2025-07-21. 2019.

[11] Answerdot AI. *ModernBERT*. https://huggingface.co/answerdotai/ModernBERT-base. Accessed: 2025-07-21. 2021.

[12] Pengcheng He, Jianfeng Gao, and Weizhu Chen. *DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing*. 2023. arXiv: 2111.09543 [cs.CL]. URL: https://arxiv.org/abs/2111.09543.

[13] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. *BERTweet: A pre-trained language model for English Tweets*. 2020. arXiv: 2005.10200 [cs.CL]. URL: https://arxiv.org/abs/2005.10200.

[14] BorisN. *EmotionBERT*. https://huggingface.co/borisn70/bert-43-multilabel-emotion-detection. Accessed: 2025-07-21. 2023.

[15] Tianyi Zhang, Xiang Zhao, and Yann LeCun. "Revisiting Few-sample BERT Fine-tuning". In: *International Conference on Learning Representations (ICLR)*. 2020.

# A Data Exploration

https://github.com/KJWesthoff/266FinalProject/blob/main/Data-Exploration_MentalManip.ipynb

# B BaseCaseModels

https://github.com/KJWesthoff/266FinalProject/tree/main/BaseCaseModels

# C 'Persuasion' Experiments

## C.1 Label distribution

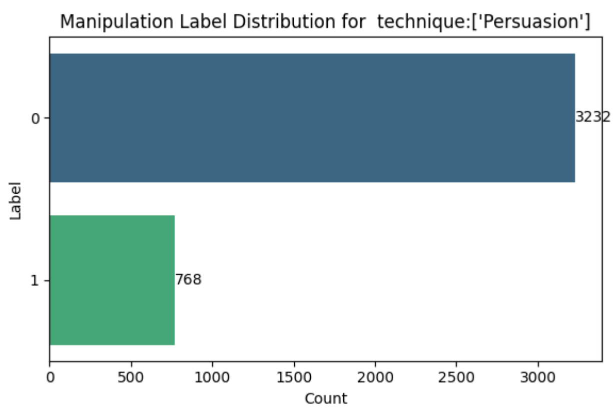


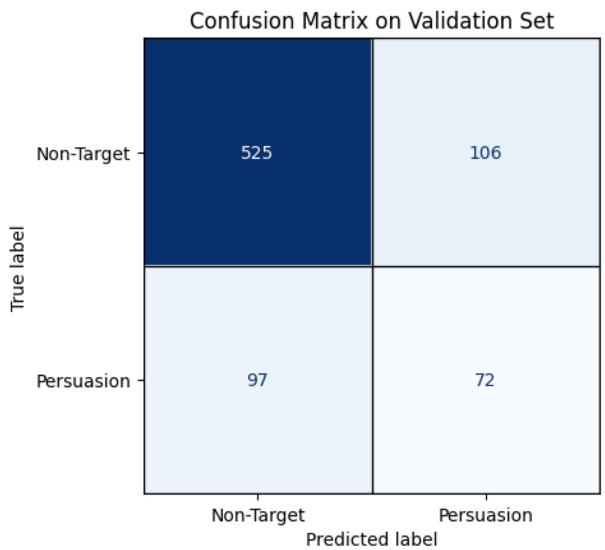

**Figure C.1:** *Label distribution with a 4.2 to 1 ratio for the 'not-persuation' to 'persuasion' labels in the* $MentalManip_{maj}$ *data set*

## C.2 Baseline Results, Un-Weighted

Notebook here: https://github.com/KJWesthoff/266FinalProject/blob/main/WeightedSkew/RoBERTa_Binary_ManipDetection_PersuasionBinary_Un-Weighted.ipynb

## C.3 Baseline Results, Weighted

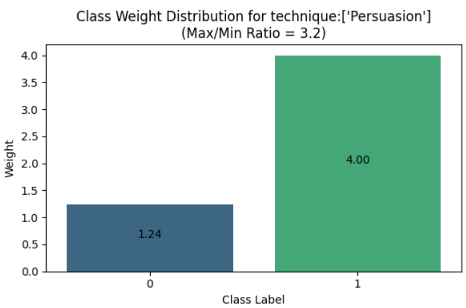
Notebooks here: https://github.com/KJWesthoff/266FinalProject/blob/main/WeightedSkew/RoBERTa_Binary_ManipDetection_PersuasionBinary_WeightedV1.ipynb https://github.com/KJWesthoff/266FinalProject/blob/main/WeightedSkew/RoBERTa_Binary_ManipDetection_PersuasionBinary_WeightedV2.ipynb https://github.com/KJWesthoff/266FinalProject/blob/main/WeightedSkew/RoBERTa_Binary_ManipDetection_PersuasionBinary_WeightedV3.ipynb

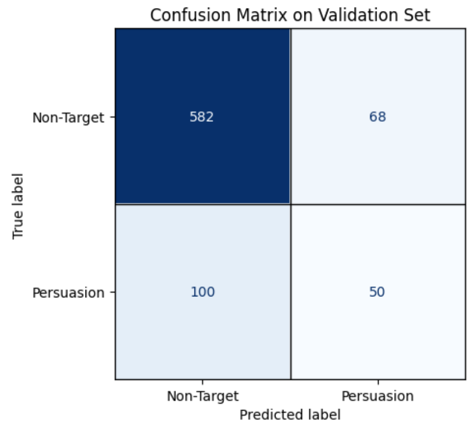



Accuracy Score: 0.75

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Non-Target | 0.84 | 0.83 | 0.84 | 631.0 |
| Persuasion | 0.4 | 0.43 | 0.41 | 169.0 |
| macro avg | 0.62 | 0.63 | 0.63 | 800.0 |
| weighted avg | 0.75 | 0.75 | 0.75 | 800.0 |

**Figure C.2:** *Confusion matrix and classification report for the Persuasion label base case model*



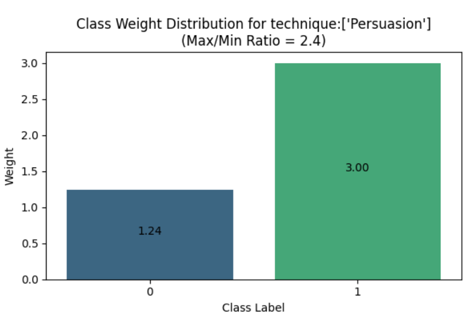



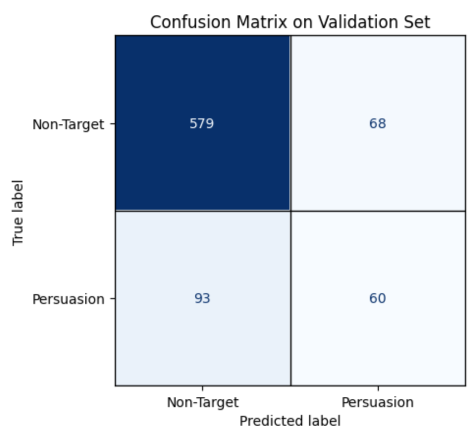

Accuracy Score: 0.79

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Non-Target | 0.85 | 0.9 | 0.87 | 650.0 |
| Persuasion | 0.42 | 0.33 | 0.37 | 150.0 |
| macro avg | 0.64 | 0.61 | 0.62 | 800.0 |
| weighted avg | 0.77 | 0.79 | 0.78 | 800.0 |

**(a)** *Distribution of weights for the Persuasion label* **(b)** *Confusion matrix and classification report for the Persuasion label base case model*

**Figure C.3:** *Results for cross entropy weight capped at 4.0*



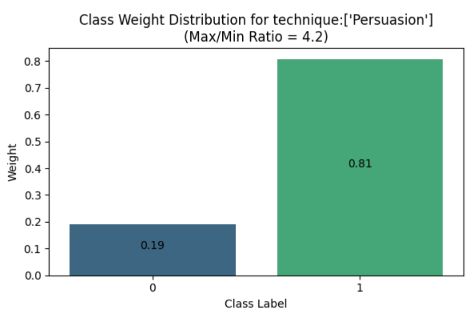



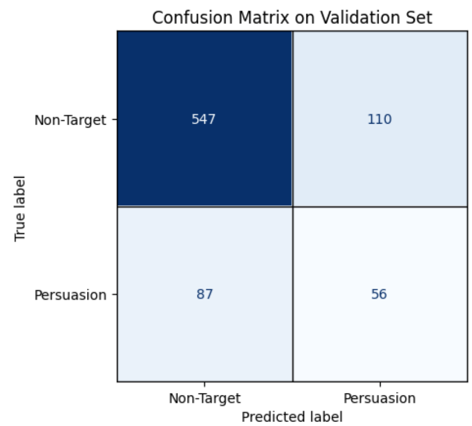

Accuracy Score: 0.8

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Non-Target | 0.86 | 0.89 | 0.88 | 647.0 |
| Persuasion | 0.47 | 0.39 | 0.43 | 153.0 |
| macro avg | 0.67 | 0.64 | 0.65 | 800.0 |
| weighted avg | 0.79 | 0.8 | 0.79 | 800.0 |

**(a)** *Distribution of weights for the Persuasion label* **(b)** *Confusion matrix and classification report for the Persuasion label base case model*

**Figure C.4:** *Results for cross entropy weight capped at 3.0*




Accuracy Score: 0.75

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Non-Target | 0.86 | 0.83 | 0.85 | 657.0 |
| Persuasion | 0.34 | 0.39 | 0.36 | 143.0 |
| macro avg | 0.6 | 0.61 | 0.6 | 800.0 |
| weighted avg | 0.77 | 0.75 | 0.76 | 800.0 |

**(a)** *Distribution of weights for the Persuasion label*

**(b)** *Confusion matrix and classification report for the Persuasion label base case model*

**Figure C.5:** *Results for normalized weighted cross-entropy loss function*