

# UZH at PAN-2023: Profiling Cryptocurrency Influencers using Ensemble of Language Models

Notebook for PAN at CLEF 2023

Abhinav Kumar, Le Hoang Minh Trinh and Afshan Anam Saeed

University of Zurich, Zurich, Switzerland

All authors contributed equally.

## Abstract

In the era of social media, the impact of social content can be massive, especially when cryptocurrency investors rely on peer advises. The crypto market has been shown to be volatile, and certain social media users have a stronger influence in the market through their social media posts than others. In this paper, we aim to classify the social media based cryptocurrency influencers into categories depicting their influence in the crypto market based on their English tweets. The task is performed under low resource setting due to limited data availability, and is done using fine tuning approaches for few shot learning. We fine tune various large language models including Bert, Roberta and Electra. We additionally fine tune a combination of models to create ensemble models to take advantages of multiple pre-trained models for our classification task. We find that the ensemble models provide better test accuracy than the single models for few shot learning tasks.

## Keywords

Natural Language Processing, Few Shot Learning, Transformers, Fine Tuning

## 1. Introduction

Cryptocurrencies, characterized by their decentralized and digital nature, have garnered immense attention from investors seeking lucrative opportunities. However, the volatile and often chaotic nature of cryptocurrency markets poses challenges for investors looking to make informed investment decisions. Unfortunately, traditional forecasting approaches that rely solely on numerical historical data are often insufficient in capturing the complex dynamics and sentiment surrounding cryptocurrencies [1].

In recent years, social media platforms have emerged as influential spaces where discussions and opinions about cryptocurrencies flourish. Tweets and online conversations about various cryptocurrencies have the power to shape public perception and drive market sentiment. This phenomenon is particularly evident during significant events, such as major price fluctuations or the introduction of new cryptocurrencies. As a result, conventional forecasting methods that overlook the impact of social media sentiment fail to capture crucial information [2].

Behavioral finance theories suggest that individuals are prone to making decisions based on

---

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18-21, 2023, Thessaloniki, Greece

✉ abhinav.kumar2@uzh.ch (A. Kumar); lehoangminh.trinh@uzh.ch (L. H. M. Trinh); afshananam.saeed@uzh.ch (A. A. Saeed)



© 2023 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



CEUR Workshop Proceedings (CEUR-WS.org)

biases and herd behavior when faced with uncertainty. In the context of cryptocurrencies, the abundance of tweets and social media discussions creates an environment where investors are susceptible to the influence of others. The so-called "hype" surrounding certain cryptocurrencies, often fueled by social media activity, can trigger herd behavior and impact investment decisions [1]. Studies have shown that cryptocurrencies valued and endorsed by experts tend to be more successful, underscoring the role of social media in shaping investor sentiment.

In volatile environments, where rapid price fluctuations and market uncertainty prevail, social media platforms become even more important for investors. Users seek advice, opinions, and insights from other social media users to gain a better understanding of market trends and potential investment opportunities. The real-time nature of social media, coupled with the ability to connect with a large community of investors, makes it an attractive source of information and a platform for sharing experiences. However, caution must be exercised, as the reliability and expertise of social media users can vary significantly.

In this work, we aim to categorize users into an influencer profile based on their English tweets using Transformer Encoder models. Since availability of social media tweets specific to Cryptocurrency influencers is limited, the goal of this project is to use this limited amount of data to create a model that is able to profile users based on their influence in the crypto market. We will begin by introducing Few-Shot Learning, Ensemble Modeling techniques, and the dataset. We would then discuss the methodology used, which includes details about the hugging face pre-trained models taken under consideration. We conclude by describing the results of our single body and two body models respectively.

## **2. Background**

### **2.1. NLP in Finance**

The field of finance has undergone a significant transformation in recent years, with traditional forecasting techniques being complemented by innovative approaches that leverage the power of natural language processing (NLP). By incorporating NLP into financial analysis, professionals can gain valuable insights into market trends, investor sentiment, and the forecasting of future trends. NLP techniques, such as sentiment analysis, event detection, trend analysis, and risk management, enhance decision-making processes and enable more accurate forecasting. In the present context, we will be using fine-tuning approaches of NLP in understanding the coherence within user tweets of a specific class, so as to identify the users significantly impacting the cryptocurrency market.

### **2.2. Few Shot Learning**

Obtaining large amounts of data for training large models is a challenging and expensive affair. Within the realm of natural language processing (NLP) tasks, the focus is on efficiently solving new tasks using only a small number of labeled examples. Recent advancements in self-supervised pre-training of transformer models, which employ language modeling objectives, have yielded remarkable achievements in learning general-purpose parameters applicable to diverse downstream NLP tasks. Nevertheless, despite the benefits of pre-training, these models

are not optimized for fine-tuning with limited supervision. Consequently, they still require substantial amounts of task-specific data to achieve satisfactory performance.

Few-shot learning tackles the challenge of training classifiers with minimal amounts of training data. This includes the extreme case of zero-shot learning, where no labeled data is available for training purposes, and few-shot classification, where classifiers are trained with only a few labeled examples per class. Few-shot learning typically relies on representing task labels in a textual format, such as their names or concise descriptions. Its applications span various domains and are widely used in image classification, sentiment classification from short text, and object recognition [3].

### **2.3. Ensemble Models**

The instances in Few Shot Learning can depend on Fine-tuning of a single pre-trained transformer model. To leverage the collective intelligence of multiple transformer models, Ensemble Deep Learning methods exist that combine the predictions of multiple individual models to obtain enhanced and more accurate results. This method allows for more robust and reliable predictions and the diversity in the models can help compensate for the weaknesses or biases of individual models, resulting in better overall performance. Due to its multiple model strength, this method is good for handling complex tasks and is a valuable technique for tasks where accurate predictions are essential.

## **3. Methodology**

### **3.1. Dataset**

Two dataset files are provided by the shared task organisers, namely, `train_text` and `train_truth`. The `train_text` dataset provided contains the following features - 'Twitter User Id', 'Text' (English tweets), and 'Tweet Id'. For every Twitter user ID, the labels are mentioned in the `train_truth` dataset. There are five classes present in the `train_truth` dataset, namely, 'no influencer', 'nano', 'micro', 'macro', and 'mega'. These labels are grouped in the order of the User Id's influence on their followers, with 'no influencer' being the least influential and 'mega' being the most influential. Consequently, it depicts how much of an influence the user has in the crypto market. There are multiple tweets given for each user, ranging from 1-10 tweets per user, and 32 users given for every class label.

### **3.2. Data pre-processing**

In this section, we pre-process the tweets which we got for training our model. We joined the `train_text` and `train_truth` dataset into a single data frame on the 'User Ids' and removed the 'Tweet Ids' from the resulting data frame, as the identifier of each tweet was not needed for our approach.

Twitter dataset underwent the following process:

- Noise data removal
- Short tweet removal

1) *Noise data removal*: Here noise means words or any text that doesn't add any relevance to the classification task. Based on this reasoning, we removed URLs present in the tweets as they are mostly short links or dead links. User mentions starting with '@' are also removed from the tweets.

2) *Short tweet removal*: Short tweets often lack sufficient information, serving as noise in datasets. Removing them enhances data quality by preserving more relevant information for processing. Some short tweets are merely retweets or emotional responses, lacking substantial content. The limited information in these tweets poses a challenge for models to discern meaningful patterns. We remove any tweet that is shorter than five words.

These features are input into various models and outputs are used to validate the models. In our experiment, we found two body ensemble model more accurate than other and in rest of the article we will focus on that. First we will talk about pre-trained models that we used and then we discuss how we combined the output of these models to get the desired result.

### 3.3. Problem Modeling

Our initial proposal for the given Few-Shot classification problem is to make use of standard fine-tuning pre-trained language models individually. Thus, we use pre-trained Large Language Models (LLMs) on tweets or for general text classification. Most of these models have more than 350M parameters and the evaluation was done on a subset of the dataset, which we took to be the test dataset. The models utilised are: Twitter RoBERTa Large [4], RoBERTa Large [5], Electra Large [6], BERTweet Large [7] finetuned on TweetNER7 dataset [8, 9]. For simplicity, this last model will be referred to as 'BERTweet Large' throughout the rest of this paper.

To evaluate the fine-tuning results on the test data, we used two strategies—referred to as the Concatenation Method and the Single Tweet Classification Method—each gauged on accuracy and F1 score metrics. Additionally, to prevent overfitting, early stopping was employed.

We used two approaches for evaluating our fine-tuning results on the test data. Both these approaches were evaluated based on the accuracy and F1 score. Early stopping was added to avoid over-fitting:

**Concatenation Method:** For a given user, we concatenate their tweets up to a maximum token size of 512. Each model then classifies these concatenated tweets into their corresponding labels. We ensure the tweets are separated by a model-specific separation token (e.g., '[SEP]' for BERT and '</s>' for RoBERTa).

**Single Tweet Classification Method:** This strategy involves each model classifying tweets individually, grouping them based on user id. To determine a user's label, we use majority voting among the classified tweets. In case of a tie, we extract the final label by averaging the probabilities.

From our preliminary analysis, it became apparent that the performance of an individually fine-tuned model left room for improvement. In addition, we observed that the Concatenation Method consistently outperformed the Single Tweet Classification Method. With these findings in mind, we sought to further enhance our performance and thus, we formulated a novel approach, which we refer to as the "Two Body Model".

### 3.4. Pretrained Models Used

#### TwHIN-BERT

TwHIN-BERT [10] stands for Twitter Heterogeneous Information Network BERT. It's a pre-trained language model built on BERT's architecture, trained on 7 billion multilingual tweets. Enhanced with social media-specific features, it is highly effective in capturing the nuances of tweets. The dataset used for training is of high quality, having been thoroughly preprocessed to remove noisy or irrelevant tweets.

#### Twitter RoBERTa Large

The Twitter RoBERTa Large model [4], is a RoBERTa based language model, specifically fine-tuned for Twitter data. Initially trained on a corpus of 90 million tweets, the model has since been continuously updated on a rapidly growing Twitter dataset, reaching a total of 154 million tweets by the time of our usage. This model's specialization and ongoing adaptation make it particularly effective at understanding the dynamic nature of discourse on Twitter. Its ability to stay current is especially valuable in fast-evolving fields such as cryptocurrencies.

#### DistilBERT

DistilBERT, a streamlined variant of BERT by Hugging Face [11], has 40% fewer parameters, thus offering speed and efficiency benefits. Despite its smaller size, it retains 95% of BERT's performance on certain benchmarks. Due to its balance between performance and computational needs, DistilBERT serves as a reliable baseline for assessing larger language models, particularly in resource-constrained environments.

#### ELECTRA-Small

ELECTRA-Small is a compact version of the ELECTRA model [6]. Its novel two-step approach enables efficient data use during training while its smaller size leads to less computational requirement. Despite its compactness, ELECTRA-Small maintains competitive performance, making it an excellent baseline for evaluating the efficiency of larger transformer models.

### 3.5. Two Body Model

In our methodology, we adopt a novel approach named the "Two Body Model". This approach involves extracting the embeddings of the CLS tokens from the last hidden layer of a pair of Transformer Encoder Models. We experimented with two sets of pairs: a pair of smaller models (Electra Small + DistilBERT) for computational efficiency, and a pair of larger models (TwHIN-BERT + Twitter RoBERTa Large), which bear the advantage of being pre-trained on Twitter data. The larger models, besides being pre-trained, also possess a more significant number of parameters. This feature allows them to capture more complex and subtle signals, leading to potentially richer and more insightful embeddings.

Once the embeddings are extracted, they are concatenated and fed into a standard Multi-Layer Perceptron that employs the Gaussian Error Linear Unit (GELU) [12] activation function.

The motivation behind GELU is to bridge stochastic regularizers, such as dropout, with non-linearities, i.e., activation functions. The output of this model provides the classification of the level of influence of the tweet’s author based on a concatenation of their tweets (using the Concatenation Method).

The primary design of this configuration is to harness the unique strengths of each component model within the pairs. The ultimate goal is to optimize performance for our specific task, which is Twitter data classification. Through this methodology, we capitalize on the specialized capabilities of the selected models, thereby ensuring that our approach is well-equipped to handle the intricacies and subtleties of Twitter discourse.

### **3.6. Head First Fine-tuning (HeFit)**

Within our experiments, we adapted a recent alternative two stage fine-tuning procedure by Head-First Fine-Tuning (HeFiT) [13] which demonstrated increase ability of Language Models to adapt to the Twitter Domain during fine-tuning.

The HeFiT approach is a two-stage process designed to gradually adapt the model to the specific task at hand. In the first stage, only the parameters of the new classification head are updated, keeping the pre-trained transformer encoder parameters frozen. This allows the new classification head to learn to make predictions using the existing representations produced by the pre-trained model. After this stage, which in our case lasted for 3 epochs, all model parameters are unfrozen and updated during training. This allows for a more nuanced adaptation of the model to the specific characteristics of the task and the data, as the encoders bodies can now adjust its representations to better suit the classification problem. An additional benefit we found with this approach was selected is signs of more stabile training and consistent performance improvements when the labeled data is scarce.

### **3.7. Experimental Setup**

We incorporated a gradient accumulation technique to deal with the constraints of GPU memory. This technique allows us to effectively increase the batch size without exceeding the memory capacity of our GPU. Specifically, we set the mini-batch size to 1 and used an accumulation step of 16, effectively simulating a batch size of 16. This was done to ensure that our model, which has a significant number of parameters, could be efficiently trained on a T4 GPU which has only 16 GB of memory. For updating network weights, gradient descent is used with Adam optimisation [14]. We have initialized learning rate with value of 0.000003 which get updated while training.

The entire fine-tuning process was carried out for a total of 20 epochs. As a result of this procedure, our model was not only able to leverage the power of two pre-trained language models but also effectively adapt to the specific characteristics of Twitter text data, demonstrating the efficacy of our Two Body Model approach.

## 4. Result

### 4.1. Standard Fine-tuning

Our evaluation results from the two fine-tuning strategies—Concatenation Method and Single Tweet Classification Method—are presented in Tables 1 and 2, respectively. From these results, it's evident that the Concatenation Method yields higher accuracy and F1 scores for the BERT and RoBERTa Models. Conversely, the Electra model demonstrates superior performance when using the Single Tweet Classification Method, achieving the highest accuracy score of 0.60 amongst all models. However, the overall accuracy and F1 scores gleaned from fine-tuning these pre-trained models are less than optimal. This may be attributed to the limited size of our labeled dataset, as well as the relative low number of parameters in these models, which can impede their capability to accurately classify labels.

**Table 1**

Evaluation of fine-tuning pre-trained models using the Concatenation Method on the Twitter dataset.

Model	Accuracy	F1-Score
Twitter RoBERTa <i>Large</i>	0.533	0.48
RoBERTa <i>Large</i>	0.467	0.376
Electra <i>Large</i>	0.267	0.195
BERTweet <i>Large</i>	0.467	0.448

**Table 2**

Evaluation of fine-tuning pre-trained models using the Single Tweet Classification Method on the Twitter dataset.

Model	Accuracy	F1-Score
Twitter RoBERTa <i>Large</i>	0.333	0.271
RoBERTa <i>Large</i>	0.40	0.361
Electra <i>Large</i>	0.60	0.541
BERTweet <i>Large</i>	0.40	0.371

### 4.2. Two-body Models

We used the Two-body Models method to accomplish the classification task to take advantage of multiple models at the same time. In our project, we worked on two combinations of models. Model 1 includes a combination of DistilBERT and ELECTRA small. Model 2 includes Twitter RoBERTa Large and TwHIN-BERT. The ensemble models underwent training for a total of 20 epochs. Evaluation of the two models on test dataset is reported in Table 3.

We see that Model 2 performed better than Model 1. Upon the completion of the final epoch, Model 2 yielded a training loss of 0.242 and achieved a training accuracy of 91.7%. Further validation of the model on a separate validation set resulted in a validation loss of 1.18 and a validation accuracy of 66.7%.



**Table 3**

Accuracy and F1 Macro scores for the two body models on test dataset.

Ensemble	Accuracy	F1-Score Macro
Model 1 (DistilBERT + ELECTRA <i>Small</i> )	0.60	0.54
Model 2 (TwHIN-BERT + Twitter RoBERTa <i>Large</i> )	0.67	0.63

A detailed performance evaluation of the TwHIN BERT and Twitter RoBERTa Large was carried out by examining precision, recall, and f1-score metrics across different classes, as seen in Table 4.

**Table 4**

Classification Report for Model 2 (TwHIN-BERT and Twitter RoBERTa Large)

Label	Precision	Recall	F1-Score
No Influencer	1.00	0.33	0.50
Nano	0.67	0.67	0.67
Micro	0.60	1.00	0.75
Macro	1.00	0.33	0.50
Mega	0.60	1.00	0.75

The 'no influencer' category saw a perfect precision of 1.00, indicating that when the model predicted a 'no influencer', it was always correct. However, the model only correctly identified 33% of the actual 'no influencer' instances, suggesting room for improvement in recall. Consequently, the f1-score, which balances both precision and recall, stood at 0.50 for this category. The model demonstrated a balanced performance for the 'nano' category, achieving both precision and recall of 0.67. This resulted in an f1-score of 0.67. For the 'micro' category, the model had a lower precision of 0.60, implying that 40% of the 'micro' predictions were incorrect. Nevertheless, it showed a perfect recall of 1.00, correctly identifying all actual 'micro' instances. This resulted in a relatively high f1-score of 0.75. The 'macro' category exhibited the same pattern as the 'no influencer' class, with a precision of 1.00, but a lower recall of 0.33, leading to an f1-score of 0.50. Finally, the 'mega' category mirrored the 'micro' class performance with a precision of 0.60, a perfect recall of 1.00, and consequently, an f1-score of 0.75.

The confusion matrix (Table 5) provides an in-depth look into the model's predictions. It shows that all instances of 'mega' and 'micro' categories were predicted correctly. On the contrary, in the 'macro' and 'no influencer' categories, the model correctly classified only one out of three instances, misclassifying the remaining ones. The 'nano' category saw two correct predictions, with one instance misclassified.

In the shared-task official result, Model 2 reported a Macro F1 score of 50.21. Our model was able to achieve a commendable performance, demonstrating its ability to effectively leverage the unique strengths of the TwHIN-BERT and Twitter RoBERTa Large models in a synergistic manner. This result further underscores the effectiveness of our novel approach to ensemble modeling and the potential of the HeFiT method.



**Table 5**

Confusion Matrix for the two body model (TwHIN-BERT and Twitter RoBERTa Large)

	No Influencer	Nano	Micro	Macro	Mega
No Influencer	1	1	1	0	0
Nano	0	2	1	0	0
Micro	0	0	3	0	0
Macro	0	0	0	1	2
Mega	0	0	0	0	3

## 5. Conclusion

In conclusion, we find from the accuracy and F1 scores that the fine-tuned two-body models are better models than single fine-tuned models. Amongst the single models, we find that Electra gave us significant performance as compared to the other models when using the majority voting approach to classify users based on their tweets. Amongst the two-body models, we obtained the best accuracy with the TwHIN BERT and Twitter RoBERTa large combination. While the model showed robust performance for certain categories, the performance was lower for others. These observations suggest possible avenues for further refinement of the model. Specifically, efforts should be made to improve the recall for the 'no influencer' and 'macro' categories and the precision for the 'micro' and 'mega' categories. Addressing these issues will likely lead to an improvement in the overall accuracy as well as individual precision and recall metrics for each class in future iterations of the model.

## 6. Future Work

For the future work, we suggest using prompting methods on the fine tuned single and two body models to check for increase in accuracy scores. Manual Prompting methods did not give us good results on the pre-trained models and thus it would be interesting to see the improvements. Additionally, higher parameter models like t5 can be worked with individually or in ensembles to make a better classification model.

## Acknowledgments

Our deepest appreciation goes to the Department of Computational Linguistics at the University of Zurich, whose provision of essential technical infrastructure was instrumental in our completion of the task. We are equally grateful to Simon Clematide and Andrianos Michail for their insightful advice and technical recommendations, which significantly enriched our work.

## References

- [1] R. Sawhney, M. Thakkar, R. Soun, A. Neerkaje, V. Sharma, D. Guhathakurta, S. Chava, Tweet based reach aware temporal attention network for NFT valuation, in: Findings

- of the Association for Computational Linguistics: EMNLP 2022, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 6321–6332. URL: <https://aclanthology.org/2022.findings-emnlp.471>.
- [2] R. Sawhney, A. Wadhwa, S. Agarwal, R. R. Shah, FAST: Financial news and tweet based time aware network for stock trading, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Association for Computational Linguistics, Online, 2021, pp. 2164–2175. URL: <https://aclanthology.org/2021.eacl-main.185>. doi:10.18653/v1/2021.eacl-main.185.
  - [3] M. Chinea-Rios, T. Müller, G. L. D. la Peña Sarracén, F. Rangel, M. Franco-Salvador, Zero and few-shot learning for author profiling, 2022. [arXiv:2204.10543](https://arxiv.org/abs/2204.10543).
  - [4] D. Loureiro, F. Barbieri, L. Neves, L. E. Anke, J. Camacho-Collados, Timelms: Diachronic language models from twitter, 2022. [arXiv:2202.03829](https://arxiv.org/abs/2202.03829).
  - [5] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, [arXiv preprint arXiv:1907.11692](https://arxiv.org/abs/1907.11692) (2019).
  - [6] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, Electra: Pre-training text encoders as discriminators rather than generators, 2020. [arXiv:2003.10555](https://arxiv.org/abs/2003.10555).
  - [7] D. Q. Nguyen, T. Vu, A. T. Nguyen, BERTweet: A pre-trained language model for English Tweets, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2020, pp. 9–14.
  - [8] A. Ushio, F. Barbieri, V. Sousa, L. Neves, J. Camacho-Collados, Named entity recognition in Twitter: A dataset and analysis on short-term temporal shifts, in: Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online only, 2022, pp. 309–319. URL: <https://aclanthology.org/2022.aacl-main.25>.
  - [9] A. Ushio, J. Camacho-Collados, T-NER: An all-round python library for transformer-based named entity recognition, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Online, 2021, pp. 53–62. URL: <https://aclanthology.org/2021.eacl-demos.7>. doi:10.18653/v1/2021.eacl-demos.7.
  - [10] X. Zhang, Y. Malkov, O. Florez, S. Park, B. McWilliams, J. Han, A. El-Kishky, Twhin-bert: A socially-enriched pre-trained language model for multilingual tweet representations, 2022. [arXiv:2209.07562](https://arxiv.org/abs/2209.07562).
  - [11] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020. [arXiv:1910.01108](https://arxiv.org/abs/1910.01108).
  - [12] D. Hendrycks, K. Gimpel, Gaussian error linear units (gelus), [arXiv: Learning](https://arxiv.org/abs/1606.05908) (2016).
  - [13] A. Michail, S. Konstantinou, S. Clematide, Uzh\_clyp at semeval-2023 task 9: Head-first fine-tuning and chatgpt data generation for cross-lingual learning in tweet intimacy prediction, [arXiv preprint arXiv:2303.01194](https://arxiv.org/abs/2303.01194) (2023).
  - [14] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).