Profiling Cryptocurrency Influencers with Few-shot Learning

Notebook for the PAN Lab at CLEF 2023

Isabel Ferri-Molla[1], Jaume Santamaria-Jorda[1]

[1]*Universitat Politècnica de Valencia, Camí de Vera, s/n, 46022 València, Valencia*

Abstract

In this paper, we describe our systems for participating in the "Profiling Cryptocurrency Influencers with Few-shot Learning" task shared on PAN 2023. This work focuses on profiling cryptocurrency influencers from limited data obtained from social networks. We employ sparse data learning techniques to classify cryptocurrency influencers into different categories. During the work, different subtasks will be tackled. On the one hand, influencers will be classified according to their number of followers. On the other hand, influencers will be classified by their interests. Finally, in the third subtask, the classification will be based on the influencer's intent. Our approach is to compare the performance of statistical models and pre-trained linguistic models, taking into account the limitations of the data.

Our approach is to compare the performance of statistical models and pre-trained linguistic models, taking into account the limitations of the data. Furthermore, we will focus on an in-depth exploration of the best parameters that can be used in the training process of the selected model to obtain the best possible metrics.

Experimental results show that the pre-trained models in all tasks obtain better global metrics even with the poor amounts of data available.

Keywords

author profiling, cryptocurrency influencers, language models

1. Introduction

The interest in cryptocurrencies has experienced a significant surge in recent years [1]. With their decentralized nature and independence from any authority, various cryptocurrency projects have gained popularity among the general public. This rise has given birth to influencers who propagate their viewpoints on social media platforms. Consequently, the profiling of cryptocurrency influencers has become a topic of increasing interest due to the substantial influence they can wield over investments and the overall market. Identifying and understanding the characteristics of these influencers can provide valuable insights for investors and companies involved in cryptocurrency.

Presently, a large segment of the population spends a considerable amount of time on social networks, particularly platforms that allow people to post short messages, such as Twitter, have gained prominent popularity. Within this social media landscape, it is evident that some users possess greater influence than others. Certain individuals' popularity can be so significant

that their opinions and messages have the power to shape the views of other users. These influential users amass a substantial following and exert a wide-ranging impact on online social interactions. Their tweets can reach diverse audiences and stimulate discussions and debates on the topics they address. Therefore, it is crucial to determine if there exists a relationship between users' level of influence and the type of tweets they post.

In this paper, we undertake the task of profiling cryptocurrency influencers using a dataset with limited data. Our work revolves around the shared task titled "Profiling Cryptocurrency Influencers with Few-Shot Learning at PAN 2023 [2], which falls within the PAN 2023 lab on digital text forensics and stylometry [3]. This study addresses the challenge of classifying cryptocurrency influencers into different categories by employing few-shot learning. This methodology proves particularly valuable when the available dataset is limited, and we aim to generalize knowledge to new samples. Our approach relies on applying machine learning techniques and comparing the performance of statistical models and pre-trained language models in this specific case. We explore various parameter combinations of the latter to identify the ones that yield superior accuracy and generalization, considering the data constraints we encounter.

2. Related word

This section provides an overview of similar solutions adopted in related problems within the field of author profiling. Author profiling focuses on analyzing and extracting key characteristics of an author based on their linguistic usage and style in text. Techniques employed in this field include natural language processing (NLP), deep learning (DL), and data analytics.

With the increasing popularity of social networking, author profiling has found significant application in these platforms. Various areas within author profiling in social networks are dedicated to predicting attributes of authors, such as gender [4, 5], age [4, 6], or personality [6, 7].

A comprehensive review of technologies used in author profiling can be found in [8].

In previous years, statistical models were widely employed in author profiling, as evidenced in [9], where techniques like Decision Trees, Random Forest, and Support Vector Machines (SVM) are utilized to discern demographic and psychometric traits based on English emails. Another notable example can be found in [10], where statistical models were utilized to profile gender and age from both English and Spanish texts.

Nevertheless, there has been a noticeable shift towards deep learning techniques, specifically the utilization of large language models (LLMs). These techniques have gained significant momentum and popularity in recent times. Moreover, these approaches have exhibited promising metrics, further reinforcing their appeal and potential, as evidenced in [11]. Additionally, the utilization of multi-model ensembles, as exemplified in [12], has become a popular approach.

In [13], a transformer-based approach is employed for author profiling, utilizing vector representations of contextualized words and hand-crafted features. This approach incorporates a self-attention mechanism and a novel coding technique that integrates stylistic, thematic, and personal information of the author. Another innovative approach explored in this field is the use of Convolutional Neural Networks (cnns), as demonstrated in [14].

When specifically considering author profiling in tweets, several studies have been conducted. For instance, [15] applies a product-based fusion strategy to combine encoded text representations from BERT_base and image features from EfficientNet. Similarly, [16] investigates the authorship of tweets related to COVID-19 in Portuguese. Further examples can be found in [17], where language aggressiveness is detected in Spanish tweets using diverse approaches such as Bag of Terms, Second Order Attributes representation, Convolutional Neural Networks, and Ensemble of N-grams.

3. Proposal approaches

The task we conducted our work for consists of three subtasks, the first subtask is about low-resource influencer profiling, the second one pertains to low-resource influencer interest identification and finally, the third subtask deals with low-resource influencer intent identification.

In the first subtask the objective was to profile influencers among 5 different categories based on their number of followers. The categories were "null", "nano", "micro", "macro" and "mega" influencers. To carry out this objective, a dataset of 160 tweeters with a list of a maximum of 10 English-language tweets was used as a training dataset. In addition, there was a truth file with the corresponding tag class for each of the tweeters.

On the other hand in the second task, the aim is to classify tweets into 5 possible areas of interest, which are "technical information", "price update", "trading matters", "gaming", and "other". The provided dataset consists on 64 tweets per label with one tweet each, all of them in English. It was accompanied by a truth file with the corresponding tag class for each user.

Finally, regarding the third subtask the data followed a similar format as in the previous task. However, the goal was to classify the influencer into one of the following categories: "subjective opinion", "financial information", "advertising", or "announcement". The given dataset was similar to the previous one, same size and characteristics, but with 4 labels.

The shared-task comprises three distinct subtasks, each focusing on low-resource influencer profiling. The first subtask involves categorizing influencers into five different categories based on their number of followers, namely "null," "nano," "micro," "macro," and "mega" influencers. To achieve this objective, a training dataset consisting of 160 tweeters was utilized, with each tweeter having a maximum of 10 English-language tweets. A corresponding truth file was provided, containing the tag class for each tweeter.

Moving on to the second subtask, the objective is to classify tweets into five possible areas of interest: "technical information," "price update," "trading matters," "gaming," and "other." The dataset provided for this task consisted of 64 tweets per label, with a single tweet per user written in English. Similar to the first subtask, a truth file accompanied the dataset, indicating the corresponding tag class for each user.

Lastly, in the third subtask, the dataset followed a similar format as the previous task. However, the goal was to classify the influencer into one of the following four categories: "subjective opinion," "financial information," "advertising," or "announcement".

In relation to subtask 1, we experimented with two different approaches due to the varying number of tweets assigned to each user. Initially, we attempted to profile each influencer by

merging all their tweets into a single string, which served as input for our model. However, this approach yielded poor results during our tests. Consequently, we pursued a second approach. In this alternative approach, we split the list of tweets corresponding to each tweeter so that each tweet was individually associated with the tweeter's category. By doing so, we determined the class mode assigned to all tweets from the same influencer, and this label was then assigned to the user.

The second approach proved to be more successful in achieving desirable results for subtask 1. As a result, we directly utilized this approach for subtask 2 and subtask 3.

Throughout our experimentation, we explored various models and solutions for the classification task. Additionally, we aimed to compare these models and evaluate their performance, taking into consideration potential variations in accuracy based on the train-test partition. To achieve this, we implemented a 5-fold cross-validation technique, which allowed us to obtain f1 and accuracy scores for each model. We strived to maintain balanced partitions during cross-validation, ensuring that samples from the same user did not appear in different partitions and aiming to have equal representation of the classes.

4. Experimental setup

This section presents the experimentation conducted for both approaches in subtask 1, as well as the experiments carried out for subtask 2 and 3.

Regarding subtask 1, each approach involved the utilization of two different methods: statistical methods and language models (LM) specifically pre-trained for the task at hand.

For the LM approach, we used tensorflow [18], to finetune and evaluate the performance of some hugging-face models, specifically the BERT-base-uncased model [19], BERTweet-base (a BERT model fine-tuned for English tweets) [20], and a RoBERTa model fine-tuned specifically for English tweets [21].

On the other hand, for the statistical approach, we employed various models from the scikit-learn library [22]. Specifically, we conducted experiments with Support Vector Machines (SVM) [23], K-means clustering [24], Perceptron [25], and logistic regression [26]. Tokenization was performed for the statistical models, wherein special characters such as @, #, etc. were replaced with corresponding keywords.

Table 1 demonstrates the outcomes obtained using the first approach. It reveals that superior results were achieved through the application of statistical methods, with logistic regression yielding the highest macro F1 score, closely followed by Support Vector Machines (SVM). Among the fine-tuned methods, the BERT-base-uncased classifier emerged as the top-performing model. These findings highlight the efficacy of logistic regression and SVM in the context of subtask 1, while also showcasing the competitive performance of the BERT-base-uncased classifier among the fine-tuned methods.

On the other hand, Table 2 presents the results obtained using the second approach. It is evident that, overall, higher metrics were achieved for all the models compared to the first approach. Notably, the finest outcomes were attained through the fine-tuning of the BERT-base-uncased model. Conversely, the statistical models exhibited slightly lower F1 scores in this case. These findings highlight the superior performance of the BERT-based approach in subtask 1 of

Table 1

Accuracy of the different tests performed with the first approach in Subtask 1

Model tested	Accuracy	MacroF1
K-means (k=5)	0.18	0.11
Logistic Regression	**0.49**	**0.50**
Perceptron (tol=1e-3)	0.45	0.43
SVM (tol=1e-3)	0.46	0.47
Fine-tuned bert-base-uncased	0.50	0.44
Fine-tuned vinai/bertweet-base	0.47	0.37

influencer profiling, further emphasizing the potential of fine-tuned language models in this domain.

Table 2

Accuracy of the different tests performed with the second approach in Subtask 1

Model tested	Accuracy	MacroF1
K-means (k=5)	0.25	0.12
Logistic Regresion	0.56	0.57
Perceptron (tol=1e-3)	0.51	0.52
SVM (tol=1e-3, random_state=0)	0.57	0.57
Fine-tuned bert-base-uncased	**0.61**	**0.59**
Fine-tuned tner/twitter-roberta-base-dec2021-tweetner7-random	0.57	0.57
Fine-tuned vinai/bertweet-base	0.56	0.55

After conducting multiple tests, we made a decision to explore a different approach inspired by the existing literature. Our main goal was to utilize Convolutional Neural Networks (CNNs) [27] for author classification, as CNNs have demonstrated their effectiveness in capturing local patterns and extracting relevant features.

To begin, we partitioned the data using the second approach described previously, then we preprocessed and normalized the text data. This involved removing HTML tags, normalizing characters, converting text to lowercase, and applying other necessary transformations. Subsequently, we performed tokenization, padding, and feature extraction to prepare the data for the CNN.

The CNN architecture incorporated Conv1D layers, which utilized filters to capture local patterns and extract word features from the embedded representations of the tweets. To optimize the performance of the CNN, we conducted experiments to determine the best parameters. After thorough exploration, we selected 180 epochs, a batch size of 128, and an embed_size of 300, Upon evaluating the model using the F1 score metric, we obtained a value of 0.45, which did not rank among the top-performing models.

As the results with the CNN were not as expected, we reverted to the approach of using pre-trained LLMs, in this case, we created an ensemble of LM using the BERT base, RoBERTa and BERTweet fine-tuned models explained above. To do so, we first pre-trained the different LMs with the Subtask 1 data, previously separated following the approach 2. Then, to classify a

new sample, we combined the prediction of the 3 models so that the class finally predicted is the mode of the predictions of all the LLMs used in the ensemble.

Although the ensemble was not a bad approach and good results were obtained, it is true that there was quite a large difference between the individual BERT base uncased and BERTweet metrics, so we wanted to test whether the results obtained with the individual BERT model by exploring its training parameters could be better than those we obtained with the assembly. We found the DistilBERT [28] variant, which is a distilled version of the BERT model, lighter and faster, but maintaining its understanding capabilities.

In order to try to improve the results of the ensemble, we tested different parameters when training DistilBERT. The initial parameters used in previous tests of BERT are based on those recommended by [19], but adapted to the size of the task. These were a learning rate of 2e-5, 6 epochs and a batch size of 16. After testing these parameters, an exhaustive exploration was carried out to find the best possible combination, from the learning rate and bach size to the the number of epochs, limiting their values to specific ranges. As results were obtained, we did different iterations in which we explored different parameters ranges to narrow down the error rate. After testing various combinations, we determined that the optimal parameters for this approach were a learning rate of 5e-5, 3 epochs, and a batch size of 4, with this parameters we obtained the best F1 score in our experiments for this task with a 0.61 of macro F1.

In order to achieve the objective of subtask 2, different models were also trained. To ensure optimal training and evaluation, a test partition was created by splitting the original training data. Cross-validation was employed by equally dividing the number of samples into folds.

As in subtask 1, we have compared different methods. The first one involved statistical models, including the ones used in subtask 1, with the addition of multilayer perceptron [29], Naive Bayes [30], Random Forest [31], and ridge classifier [32]. These models have demonstrated competitive results even with limited data. The second approach involved fine-tuning pre-trained models, which, akin to Subtask 1, demonstrated improved performance. These results can be observed in Table 3. The best parameters found for this task were also the same as for the previous one. The second approach focused on fine-tuning pre-trained models, proved to be more effective as observed in subtask 1. The results of these different models can be observed in Table 3.

In Subtask 3, we employed a similar methodology as in Subtask 2. Initially, we conducted experiments using different statistical models. However, these models did not attain the desired level of accuracy. Consequently, we focused our efforts primarily on testing and fine-tuning the DistilBERT model, as it had exhibited the most promising outcomes in previous tasks.

To evaluate the performance of the models, we compared their accuracy and F1 metrics. The results are summarized in Table 4.

The table 4 provided illustrates the accuracy and F1 scores attained by various models. Among the statistical models, accuracy scores ranged from 0.56 to 0.62, while F1 scores fell between 0.58 and 0.61. However, the pre-trained language models demonstrated superior performance compared to the statistical models, achieving accuracy scores of up to 0.83 and F1 scores of 0.84.

Table 3
Accuracy and MacroF1 of the different tests performed in Subtask 2

Model tested	Accuracy	MacroF1
KNeighbors (metric=minkowski, weight=distance)	0.35	0.36
Logistic Regression	0.42	0.43
MLP (ReLU activation, batch=64, lr=0.01, α=1e-5)	0.40	0.4
Naive Bayes	0.37	0.37
Perceptron (tol=1e-3)	0.41	0.39
RandomForest (No bootstrap, estimators=800)	0.47	0.46
Ridge (α=1.0)	0.45	0.46
SGD (loss=modified huber, penalty=l1)	0.43	0.41
SVC (C=10, kernel=rbf)	0.44	0.43
SVM (tol=1e-3)	0.43	0.42
DistilBERT fine-tuned	**0.63**	**0.63**
RobBERTa tner/twitter-roberta-base-dec2021-tweetner7-random fine-tuned	0.63	0.57

Table 4
Accuracy of the different tests performed in Subtask 3

Model tested	Accuracy	MacroF1
SVM (tol=1e-3)	0.62	0.61
Perceptron (tol=1e-3)	0.59	0.58
Logistic Regression	0.56	0.58
DistilBERT fine-tuned	**0.83**	**0.84**

5. Results

This section presents the final results obtained in TIRA[1].

In subtask 1, two different models were tested on the platform. Firstly, the ensemble of the three different language models (LM) discussed in section 4 was evaluated, achieving an F1 score of **0.45**.

Additionally, the DistilBERT model explained in section 4 was employed too. Regarding DistilBERT models after the parameter exploration, we determined that the optimal parameters for this approach were a learning rate of 5e-5, 3 epochs, and a batch size of 4. Notably, this approach outperformed the ensemble approach in TIRA, attaining a macro F1 score of **0.57**.

These results highlight the effectiveness of the DistilBERT model for low-resource influencer profiling in subtask 1, surpassing the performance of the ensemble model on the TIRA platform.

Regarding second subtask after the experimentation explained in section 4 we got the conclusion that DistilBERT fine-tuned obtained the best F1 metric so this one was the one presented in TIRA. This model achieved final results on the platform of **0.55** macro F1 score.

In Subtask 3, our experimental findings revealed that the DistilBERT fine-tuned model yielded

---

[1]TIRA is a platform for reproducible participation in shared tasks from information retrieval, natural language processing, and machine learning, where organizers can provide datasets to participants and manage their submissions. https://www.tira.io/

the most favorable outcomes that is the reason why this one whas the model tested in the platform. It consistently achieved results in the TIRA evaluation that surpassed the average performance, exhibiting a noteworthy macro F1 score of **0.61**.

Extensive experimentation was conducted for the DistilBERT models employed in both Subtasks 2 and 3 to identify the optimal parameters. Surprisingly, the best parameters obtained for both tasks were consistent: a learning rate of 5e-5, 3 epochs, and a batch size of 4, mirroring the parameters used in Subtask 1. This observation suggests that the tasks may share a certain level of similarity, leading to the convergence of optimal parameter values across them.

6. Conclusion and Future Works

In this work we have trained several models with few data in order to profile criptocurrency influencers. Throughout our experiments, we observed that fine-tuning pre-trained models generally yielded superior results than using statistical models. Specifically, for Subtask 1 we found that splitting the tweet list of each influencer and individually associating each tweet with the corresponding label proved to be a more effective approach. Although the performance improvement over statistical models was not as substantial compared to other subtasks, the fine-tuned neural models demonstrated better performance. Through an ensemble of neural models in TIRA, we achieved an F1 score of 0.45. Furthermore, after extensive parameter testing, we obtained the best results using a pre-trained DistilBERT model, achieving an F1 score of 0.57.

In relation to Subtask 2, a more pronounced disparity was observed between statistical and neural models. Notably, the best outcomes were achieved using a DistilBERT model, which yielded an F1 score of 0.55. Finally, in relation with Subtask 3, we once again experimented with both statistical and neural models. After exploring various training parameters, it was determined that a fine-tuned DistilBERT model emerged as the superior choice, resulting in an F1 metric of 0.61.

Although the best results have been obtained with pre-trained models, there is still room for improvement and it would be of interest to explore other models, as well as alternative structures and ways of assembling them Given that the statistical techniques have given a good overall result, it would be interesting to further explore them and to test with ensembles of models, as well as experiment with new ways of preprocessing data.

References

[1] R. Sawhney, S. Agarwal, V. Mittal, P. Rosso, V. Nanda, S. Chava, Cryptocurrency bubble detection: A new stock market dataset, financial task & hyperbolic models, in: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Seattle, United States, 2022, pp. 5531–5545. URL: https://aclanthology.org/2022.naacl-main.405. doi:10.18653/v1/2022.naacl-main.405.

[2] M. Chinea-Rios, I. Borrego-Obrador, M. Franco-Salvador, F. Rangel, P. Rosso, Profiling Cryptocurrency Influencers with Few shot Learning at PAN 2023, in: CLEF 2022 Labs and Workshops, Notebook Papers, 2023.

[3] J. Bevendorff, I. Borrego-Obrador, M. Chinea-Ríos, M. Franco-Salvador, M. Fröbe, A. Heini, K. Kredens, M. Mayerl, P. Pęzik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, , E. Zangerle, Overview of PAN 2023: Authorship Verification, Multi-Author Writing Style Analysis, Profiling Cryptocurrency Influencers, and Trigger Detection, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, A. G. Stefanos Vrochidis, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multi-linguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023), Lecture Notes in Computer Science, Springer, 2023.

[4] S. Goswami, S. Sarkar, M. Rustagi, Stylometric analysis of bloggers' age and gender, in: Proceedings of the International AAAI Conference on Web and Social Media, volume 3, 2009, pp. 214–217.

[5] L. Flekova, I. Gurevych, Can we hide in the web? large scale simultaneous age and gender author profiling in social media, in: CLEF 2012 Labs and Workshop, Notebook Papers, 2013.

[6] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman, et al., Personality, gender, and age in the language of social media: The open-vocabulary approach, PloS one 8 (2013) e73791.

[7] Y. Bachrach, M. Kosinski, T. Graepel, P. Kohli, D. Stillwell, Personality and patterns of facebook usage, in: Proceedings of the 4th annual ACM web science conference, 2012, pp. 24–32.

[8] J. Bevendorff, M. Chinea-Ríos, M. Franco-Salvador, A. Heini, E. Körner, K. Kredens, M. Mayerl, P. Pęzik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of pan 2023: Authorship verification, multi-author writing style analysis, profiling cryptocurrency influencers, and trigger detection, in: J. Kamps, L. Goeuriot, F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, A. Caputo (Eds.), Advances in Information Retrieval, Springer Nature Switzerland, Cham, 2023, pp. 518–526.

[9] D. Estival, T. Gaustad, S. B. Pham, W. Radford, B. Hutchinson, Author profiling for english emails, in: Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics, volume 263, Citeseer, 2007, p. 272.

[10] M. De-Arteaga, S. Jimenez, G. Duenas, S. Mancera, J. Baquero, Author profiling using corpus statistics, lexicons and stylistic features, Online Working Notes of the 10th PAN evaluation lab on uncovering plagiarism, authorship. and social misuse, CLEF (2013).

[11] M. Fabien, E. Villatoro-Tello, P. Motlicek, S. Parida, Bertaa: Bert fine-tuning for authorship attribution, in: Proceedings of the 17th International Conference on Natural Language Processing (ICON), 2020, pp. 127–137.

[12] J. P. Delmondes Neto, I. Paraboni, Multi-source bert stack ensemble for cross-domain author profiling, Expert Systems 39 (2022) e12869.

[13] R. López-Santillán, L. C. González, M. Montes-y Gómez, A. P. López-Monroy, When attention is not enough to unveil a text's author profile: Enhancing a transformer with a wide branch, Neural Computing and Applications (2023) 1–20.

[14] M. E. Aragón, A.-P. López-Monroy, A straightforward multimodal approach for author profiling, in: Proceedings of the Ninth International Conference of the CLEF Association

(CLEF 2018), 2018.

[15] C. Suman, A. Naman, S. Saha, P. Bhattacharyya, A multimodal author profiling system for tweets, IEEE Transactions on Computational Social Systems 8 (2021) 1407–1416.

[16] P. V. Brum, M. C. Teixeira, R. Miranda, R. Vimieiro, W. Meira Jr, G. L. Pappa, A characterization of portuguese tweets regarding the covid-19 pandemic, in: Anais do VIII Symposium on Knowledge Discovery, Mining and Learning, SBC, 2020, pp. 177–184.

[17] M. E. Aragón, A. P. López-Monroy, Author profiling and aggressiveness detection in spanish tweets: Mex-a3t 2018., in: IberEval@ SEPLN, 2018, pp. 134–139.

[18] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL: https://www.tensorflow.org/, software available from tensorflow.org.

[19] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[20] 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.

[21] 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.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[23] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, B. Scholkopf, Support vector machines, IEEE Intelligent Systems and their applications 13 (1998) 18–28.

[24] J. A. Hartigan, M. A. Wong, Algorithm as 136: A k-means clustering algorithm, Journal of the royal statistical society. series c (applied statistics) 28 (1979) 100–108.

[25] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain., Psychological review 65 (1958) 386.

[26] R. E. Wright, Logistic regression. (1995).

[27] S. Albawi, T. A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network, in: 2017 international conference on engineering and technology (ICET), Ieee, 2017, pp. 1–6.

[28] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller,

faster, cheaper and lighter, 2020. `arXiv:1910.01108`.

[29] M. W. Gardner, S. Dorling, Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences, Atmospheric environment 32 (1998) 2627–2636.

[30] I. Rish, et al., An empirical study of the naive bayes classifier, in: IJCAI 2001 workshop on empirical methods in artificial intelligence, volume 3, 2001, pp. 41–46.

[31] L. Breiman, Random forests, Machine learning 45 (2001) 5–32.

[32] J. He, L. Ding, L. Jiang, L. Ma, Kernel ridge regression classification, in: 2014 International Joint Conference on Neural Networks (IJCNN), IEEE, 2014, pp. 2263–2267.