Neural networks based ensemble classifier for phishing link detection

Wojciech Gałka
Institute of Computer Science / University of Rzeszów
Rzeszów, Poland *wgalka@ur.edu.pl*

Marcin Mrukowicz
Institute of Computer Science / University of Rzeszów
Rzeszów, Poland *mmrukowicz@ur.edu.pl*

Urszula Bentkowska
Institute of Computer Science / University of Rzeszów
Rzeszów, Poland *ubentkowska@ur.edu.pl*

Jan G. Bazan
Institute of Computer Science / University of Rzeszów
Rzeszów, Poland *jbazan@ur.edu.pl*

Abstract

This contribution proposes an ensemble classification model which is based on neural networks prediction models and well-known online incremental learning models. The considered neural network models belong to different families, namely long-short term memory, deep feed forward and convolutional neural networks. The incremental learning models considered are Passive Aggressive, Bernoulli Naive Bayes and Stochastic Gradient Descent Classifiers. This paper aims to develop a prediction model that reduces false positives (FP) while maintaining overall model performance. Moreover, the stability of the model over time and its ability to correctly classify phishing links, even if the concept shift occur, are under considerations. The ensemble model shows promising results, demonstrating its superiority over base models. Some proposed models significantly outperform some base models according to statistical tests.

Keywords: neural networks, online learning, aggregation functions, online incremental learning, ensemble learning

1. Introduction

Phishing is a fraud attempt where cybercriminals impersonate trusted entities to obtain sensitive user information. It remains a common tactic among cybercriminals, requiring continuous efforts for successful detection [6]. Attackers often adapt their actions to avoid detection, making traditional batch learning approach may be insufficient to this problem. A possible solution is to employ an online incremental learning approach [3, 4], [6]. A machine learning-based phishing detection tool can be customized for different users. In the context of commercial email sender company, it's crucial to use this tool effectively without sending phishing messages. Experts must verify suspected emails. Senders may claim compensation for emails wrongly blocked. The detector should minimize false positives (FPs), as these can be expensive to deal with.

This contribution aims to reduce FP while maintaining overall model performance. Another studied issue is the stability of the model over time, and especially its ability to correctly classify phishing links (even if the possible concept drift occurs). This problem was previously studied in [3] and [4]. The models denoted as w_1-w_3 in Section 2 were originally used in [6] and later adopted also in [3] and [4]. In [3] and [4] the proposed model was an ensemble model. In [3] only the well-known aggregations such as min, max, and the arithmetic mean were used to

combine the predictions of the individual models, while in [4] various families of aggregation functions and uninorms was additionally utilized. The neural networks architectures used in this contribution (denoted as f_1, f_2, l, c_1 and c_2) were originally proposed in [4], where they were considered as State of The Art (SOTA) models, but where not a part of the ensemble. In both [3] and [4] there was applied a threshold moving strategy, which main goal was to tune the model to obtain desired minimal TPR. Contrary in this contribution the models are not tuned in this fashion.

The main novelty of this paper is to use an ensemble model based on both online incremental learning models and neural network models. Moreover, other families of aggregation functions are applied and the classification algorithm is modified (as stated before the models are not tuned to obtain the desired minimal TPR). Finally, in this contribution, the stability of the considered model with special attention to the concept drift problem is studied.

2. Methodology

Passive aggressive classifier (denoted as w_1), online Bernoulli Naive Bayes (denoted as w_2) and stochastic gradient descent classifier (denoted as w_3) were considered as base models and were previously used in [3, 4], [6]. They are both popular online learning models and a popular choice for phishing link detection [4]. For w_1-w_3 scikit-learn implementations were applied with the default values of all hyperparameters.

The neural networks utilized in this study represent a popular choice in the domain of the phishing link detection [4]. They came from different families: LSTM (Long-Short Term Memory), deep feed forward neural networks (DFFNN) and CNN (Convolutional Neural Network). The f_1 is a DFFNN, which contains one single hidden layer, while f_2 is a DFFNN, which contains 3 hidden layers and one dropout layer. The c_1 is a CNN, which uses kernel of size 3 to generate 32 channels. The c_2 is a CNN, which uses kernel of size 3x3 to generate 8 channels. Both CNNs applied GELU as an activation function and contain one additional feed forward layer. The l is a LSTM, which has 20 hidden dimensions. The exact architectures of these networks are the same as in [4], where also the more detailed description about them can be found.

The proposed ensemble model consist of individual models, which are the nonempty subsets of the set $\{ w_1, w_2, w_3, f_1, f_2, l, c_1, c_2\}$ which are independently trained and their predictions are combined using aggregation functions from the families described in [1]: A_{mx}, A_{pr}, A_{qd}, A_{gm}, A_{hm}, $A_{pw}^{-0.5}$, $A_{pw}^{0.5}$, $A_{pw}^{1.5}$, $A_{pw}^{-1.5}$, A_{pw}^{3}, A_{pw}^{-3}, $A_{ex}^{-0.5}$, $A_{ex}^{0.5}$, A_{ex}^{2}, A_{ex}^{-2}, $A_{lm}^{0.5}$, $A_{lm}^{-0.5}$, $A_{lm}^{-1.5}$, $A_{lm}^{1.5}$, A_{lm}^{3}, A_{lm}^{-3}, A_{md}, A_{ol}, A_{oln}^{2}, and convex combinations $A_{ar,mn}^{0}$, $A_{ar,mn}^{0.05}$, $A_{ar,mn}^{0.01}$, $A_{ar,mn}^{1}$, $A_{gm,mn}^{0.05}$, $A_{gm,mn}^{0.1}$, $A_{gm,mn}^{1}$, $A_{pr,mn}^{0.05}$, $A_{pr,mn}^{0.1}$, $A_{pr,mn}^{1}$. The total number of combinations between base models and aggregations gives the total number of 7 904 examined proposed models. Because of the fact, that online learning paradigm was adopted, the proposed ensemble model after trained on initial data is later additionally trained on consequent data chunks. The base models also used this learning paradigm. The details of proposed ensemble model is visualized in Fig. 1.

The FreshMail dataset used in this research consists of 19 features (columns) and 2 564 973 records collected over a time span of 120 days and was previously used in papers [3, 4], where more details about it can be found. To simulate a real-time scenario, the dataset is divided into 12 sequential chunks, each representing 10 consecutive days of observations. This temporal division allows the model to mimic a streaming environment in which it receives and processes new data at regular intervals. All available features were used in the training phase.

As stated previously in the introduction, attackers usually modify their actions and therefore the characteristics of the data is changing over time. As suggested in [5] the Population Stability Index (PSI) was applied as a measure of detecting drifts (shifts) in data. PSI may be defined as a variant of the Kullback-Leibler (KL) divergence measure (relative entropy). Unlike KL

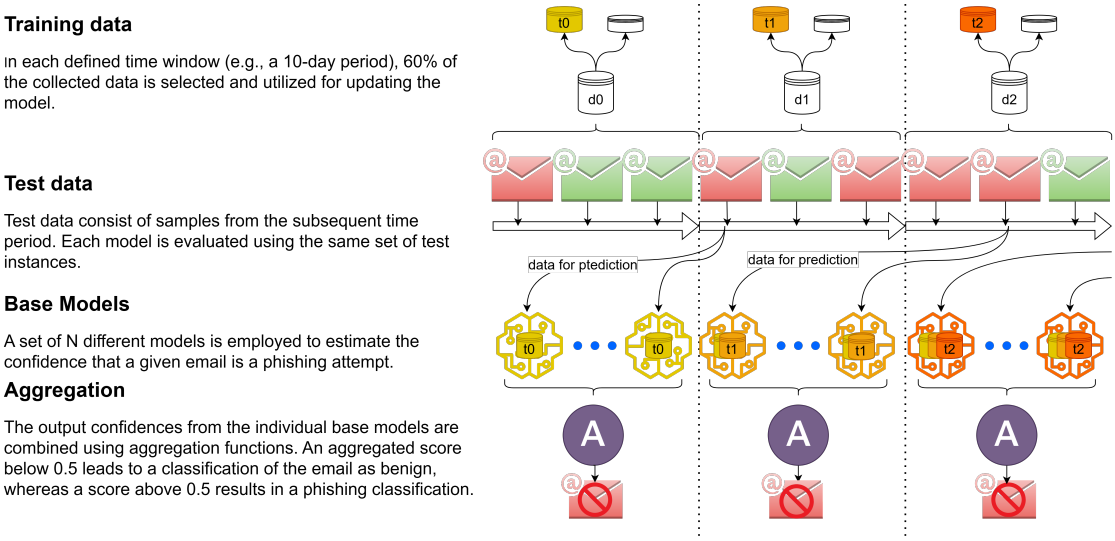

Fig. 1. Proposed model working scheme

divergence PSI is symmetrical and therefore is a distance measure. PSI has a widely accepted rule of thumb, that PSI < 0.1 is considered no significant drift, $0.1 \leq PSI \leq 0.2$ is considered a substantial divergence, and PSI > 0.2 is considered as a significant shift [5]. However, this rule is non-strict and should be understood as a guideline. It is worth noting, that concept drift is a broad term. The main categories are [7]: drift subject (class drift, covariate drift); frequency (abrupt drift, extended drift); reoccurrence, magnitude and transition (gradual, incremental drift). The PSI is a quantitative measure best suited to detecting covariate drift (when the distribution of non-class attributes changes over time). Later in the paper, the concept shift will be understood as a covariate drift.

3. Results

Among models with the lowest reported mean FP, we chose their representatives: $A_{lm}^{r=-3}(w_1, w_2, w_3, c_1, c_2, f_1, f_2, l)$ denoted as pm_1 and $A_{pr}(w_1, w_2, c_1, c_2, f_1, f_2, l)$ denoted as pm_2. The best ensemble of only neural networks is $A_{pr}(c_1, c_2, f_1, f_2, l)$ denoted as pm_3. The best ensembles of only $w_1 - w_3$ are $A_{pw}^{-3}(w_1, w_2, w_3)$ labelled pm_4 and $A_{pr}(w_1, w_2, w_3)$ labelled pm_6. The $A_{ol}(w_2, c_1, f_1, f_2, l)$ denoted as pm_5 is the model with the highest observed ROC AUC.

As shown in Table 1, the mean FP for the best proposed models is between 111 and 115, while the best neural network c_1 has a score of 438 (about 4 times greater). c_1 and f_1 are comparable, while l and f_2 are the second best. c_2 is close to w_3 and is clearly the weakest neural network model. The weak performance of w_2 could be due to its probabilistic nature and its susceptibility to strong concept shift. Every base model generates some FPs on every fold, while the best proposed models generate lower FPs (even 0). Notably, over 3 300 aggregation-based ensembles are better than the best neural network c_1.

For folds 4 and 10, the number of false positives (FPs) is growing rapidly compared to previous folds. On a smaller scale, the growth of FPs is between folds 0 and 1, and folds 2 and 3. This suggests two important concept shifts appear in the data. To verify this, the PSI was calculated between every train and test fold using 10 bins. In [2] there are listed real features names with their corresponding abbreviations. Nine features (denoted as F_0-F_6, F_8, F_9) have numerous PSI values greater than 0.2. The remaining 10 features do not show any concept shift, or it is not evident (the PSI is greater than 0.1 sometimes). We chose F_7, F_{10} and F_{11} as their representatives. In [2] the PSI values are presented for these selected features.

The amount of false phishing link detections is also determined by the amount of clear links in each fold itself (if this number is greater simply the number of false detections could be greater too). The count of clear occurrences in every test fold is listed in [2]. Fold 4 has especially many FPs for all models but its count of clear links is similar to fold 5 and is lower than fold 6 or 7, where FPs are generally less numerous. Fold 10 is one of the smallest, while reported FPs for some models are very high. Therefore the occurrence of concept drift is more probably the reason for the drop of classification quality and not the characteristic of the dataset itself.

It is evident that train fold 0 is significantly different than every test folds (even test fold 0). The difference between train and test fold 0 explains why base models are generating even hundreds or thousands of FPs on it. Interestingly, some proposed models seem to be more prone to this concept drift. In test fold 1 pm_1 and pm_4 have more FPs, despite training on similar train fold 1. The quality of the other models is generally improving, but the base models are worse than the best proposed models. The test fold 4 differs very significantly from train fold 0 and significantly on some features from folds 1-3. This leads to an outcome that some concept drift occurs on test fold 4 and all models trained on folds 0-3 drop their quality. The test fold 4, according to PSI is not dissimilar from train folds 4, 5, 6 and is only a little different than train folds 7 and 8. With train fold 9 the difference is very significant on 3 features. In other words, it may be assumed that folds 4–8 represent a similar pattern and that all models have a tendency to adapt to it and improve their quality over time, reaching a peak at fold 8. For fold 9 the base models rapidly lose quality, while the best proposed models maintain relatively high quality. Test fold 10 differs significantly from almost all train folds, particularly its predecessors, folds 8 and 9. Even train fold 10 differs from test fold 10. For fold 10 the growth of the FPs is the highest, but the best proposed models are more prone to this.

To verify if the observed differences in quality are significant we applied statistical tests. We tested only groups from Table 1. As FPs values are not normally distributed, the Kruskall-Wallis test was used to check for differences between groups. This test confirmed significant differences (p-value was $5.0 \cdot 10^{-7}$), so the post-hoc Dunn's test with Holm-Bonferroni corrections was used to identify which groups were significantly different. Between any pair of the best proposed models Dunn's test shows no difference. pm_1 is statistically different from w_1-w_3 and c_2. w_2 is worse than pm_1-pm_4. c_1, f_1, f_2, and l are not significantly different from any proposed model. The p-values are presented in [2]. We used an alpha level of 0.05 for all statistical tests.

To better understand the overall behavior of the models, the mean ROC AUC was calculated. The base models have values of approximately 0.99, with w_3 being an exception at approx. 0.95. pm_1 and pm_4 have approx. 0.74, pm_6 has approx. 0.95, while pm_2 and pm_3 have approx 0.99 and $pm_5 >$ 0.999.

4. Conclusions

pm_2 has the fewest reported FPs with relatively high ROC AUC. It is the best recommended setup. pm_1 has the fewest generated FPs but the lowest ROC AUC. At least 5 models in the ensemble are requisite to attain the highest quality. The best ensemble models combine $w_1 - w_2$ with neural networks (compare pm_2 and pm_3). A_{pr} seems to be the optimal aggregation. A_{lm} and A_{pw} decrease FPs while lowering the overall quality. A_{ol} increases ROC AUC, while increasing the number of FPs.

The ensemble of neural networks first introduced in [4] exceeds their individual classification quality. The same happens in the case of the w_1-w_3. The aspect of data changing over time was better studied than in the [3] and [4]. With the usage of the PSI measure the concept drifts was observed with high probability, and the aspect of model durability to this fact was examined. The main achievement of the proposed model is that it generates a significantly lower number of FPs than base models (considered as SOTA in this area). Moreover, when some concept

Table 1. The best ensemble models results and the base models FP

	Fold											
abb.	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Mean**
pm_1	0	149	0	151	636	82	121	84	0	0	0	111
pm_2	87	17	78	78	239	46	61	48	3	99	512	115
pm_3	123	71	105	91	568	79	143	68	19	104	827	200
pm_4	0	497	0	488	1679	2189	352	360	0	0	0	506
pm_5	580	244	324	241	892	2278	357	207	139	565	1972	709
pm_6	639	188	512	467	1656	2185	145	359	35	792	1593	779
c_1	440	195	276	260	904	344	315	192	104	493	1296	438
f_1	439	199	205	195	787	291	310	159	100	456	1768	446
l	824	307	398	304	792	2294	376	209	157	579	2147	762
f_2	824	282	336	251	1211	2314	376	203	132	304	2766	818
w_3	1838	661	2413	817	2504	2190	374	366	329	917	1671	1280
c_2	556	617	513	750	1485	3974	708	593	320	1074	5719	1483
w_1	1892	196	677	1415	3993	4098	150	3764	35	1284	30138	4331
w_2	821	2454	2186	2780	3502	8373	5810	5673	3649	7273	26501	6275

drift happens, it is more stable and more prone to rapid increase of FPs than SOTA. It was also shown that the proposed models maintain good overall quality (comparable ROC AUC to the base models).

References

[1] aggregationslib: Python implementation of arithmetic, quasi-arithmetic and other aggregation functions. `https://pypi.org/project/aggregationslib/` (2025), accessed Jun 20, 2025

[2] Appendix to isd 2025 paper. `https://wgalka.github.io/appendix-ISD-2025/` (2025), accessed Jun 20, 2025

[3] Gałka, W., Bazan, J.G., Bentkowska, U., Mrukowicz, M., Drygaś, P., Ochab, M., Suszalski, P., Obara, S.: Self-tuning framework to reduce the number of false positive instances using aggregation functions in ensemble classifier. Procedia Computer Science 246, pp. 4028–4037 (2024), 28th International Conference on Knowledge Based and Intelligent information and Engineering Systems (KES 2024)

[4] Gałka, W., Bazan, J.G., Bentkowska, U., Szwed, K., Mrukowicz, M., Drygaś, P., Zaręba, L., Szpyrka, M., Suszalski, P., Obara, S.: Aggregation-based ensemble classifier versus neural networks models for recognizing phishing attacks. IEEE Access 13, pp. 48469–48487 (2025)

[5] Kurian, J.F., Allali, M.: Detecting drifts in data streams using kullback-leibler (kl) divergence measure for data engineering applications. Journal of Data, Information and Management 6(3), pp. 207–216 (2024)

[6] Prasad, A., Chandra, S.: Phiusiil: A diverse security profile empowered phishing url detection framework based on similarity index and incremental learning. Computers & Security 136, pp. 103545 (2024)

[7] Webb, G.I., Hyde, R., Cao, H., Nguyen, H.L., Petitjean, F.: Characterizing concept drift. Data Mining and Knowledge Discovery 30(4), pp. 964–994 (2016)