Dynamic Spammer Detection Using Deep Learning with Temporal Graph Embeddings

Hemza Loucif[1]✉, Samir Akhrouf[2]

[1] *Laboratory of Informatics and its Applications, University of M'sila, M'sila 28020, Algeria*
hemza.loucif@univ-msila.dz
samir.akhrouf@univ-msila.dz

Abstract

Spammers in online social networks continuously adapt their strategies, making detection a challenging and dynamic task. While traditional machine learning models and static deep learning approaches such as CNNs achieve good performance, they often fail to capture the temporal evolution of user behavior and network interactions. In this paper, we propose a novel deep learning framework for dynamic spammer detection that combines Principal Component Analysis (PCA) for feature reduction, Convolutional Neural Networks (CNNs) for local content feature extraction, and Temporal Graph Embeddings (TGEs) to capture evolving interaction patterns over time. Unlike prior static models, our approach explicitly models the dynamics of user behavior and relational changes in the social graph. Experiments conducted on a benchmark Twitter dataset demonstrate that our hybrid PCA–CNN–TGE model significantly outperforms classical baselines as well as static hybrid models, achieving an F1-score of 94 %. The results highlight the importance of temporal graph learning for robust and adaptive spammer detection in social networks.

Keywords: *Spam, Cybersecurity, CNN, Social Networks, Temporal Graph Embeddings, PCA.*

1. Introduction

Online Social Networks (OSNs) such as X (previously Twitter[1]), Facebook, and Instagram have become central platforms for global communication, information dissemination, and public opinion formation. Billions of users worldwide interact on these platforms daily, sharing short text messages, images, videos, and hyperlinks. While OSNs have transformed social, political, and business communication, they have also become a fertile ground for malicious activities, particularly spamming. Spammers are accounts—either automated bots or compromised human users—that disseminate unsolicited, misleading, or harmful content. Their activities include phishing, fraudulent promotions, rumor spreading, and fake news campaigns. According to recent studies[2], nearly one in five Twitter accounts exhibits suspicious or spam-like behavior, posing serious threats to cybersecurity, information reliability, and user trust.

The dynamic nature of spammers makes their detection particularly challenging. Unlike conventional attackers, spammers frequently adapt their strategies to evade detection systems. They may vary the content they post, alter posting frequencies, change their interaction patterns, or manipulate their social connections. This evolutionary behavior undermines static detection systems, which typically rely on either content analysis or fixed structural features. As a result, dynamic spam detection—the ability to monitor and analyze how user behavior changes over time—has become an essential yet underexplored research direction in social network analysis.

[1] In this paper, we use the term "Twitter" when referring to datasets collected before the rebranding.

[2] https://sparktoro.com/blog/sparktoro-followerwonk-joint-twitter-analysis-19-42-of-active-accounts-are-fake-or-spam/

Traditional spam detection techniques relied on rule-based filtering or classical machine learning algorithms such as Support Vector Machines (SVMs), Random Forests, and Naïve Bayes [1], [2]. These approaches depend heavily on handcrafted features such as posting frequency, follower–friend ratios, or URL counts. While effective in controlled settings, handcrafted methods struggle to generalize across platforms and against adaptive spammers. More recently, deep learning approaches have gained traction due to their ability to automatically extract complex features from unstructured data. Convolutional Neural Networks (CNNs), in particular, have demonstrated success in learning discriminative patterns from textual and behavioral features in OSNs [3], [5].

In previous work, CNN-based architectures have been extended with dimensionality reduction techniques such as Principal Component Analysis (PCA). For example, the PCA–CNN hybrid model reduces input feature dimensionality, mitigating overfitting and improving classification accuracy [6]. Such hybrid architectures have achieved strong results in static spammer detection tasks. However, their key limitation lies in their static assumption: they treat user behavior as fixed rather than evolving, ignoring the temporal aspect of interactions in social networks.

In reality, spammers operate dynamically. Their behavior evolves as they change posting times, modify their linguistic patterns, or manipulate interaction strategies with legitimate users. Static models may initially detect such spammers but often fail when adversaries adopt new tactics. Therefore, a critical next step in spam detection research is to integrate temporal modeling into deep learning frameworks. Temporal modeling enables the system to detect spammers not only by analyzing their content and structural features but also by tracking how their behavior changes over time.

Graph-based methods offer a powerful tool for this purpose. Social networks can be naturally represented as graphs, where nodes correspond to users and edges correspond to interactions such as mentions, retweets, or replies. Recent advances in Graph Neural Networks (GNNs) [7], [8] and Temporal Graph Embeddings (TGEs) [9], [12] allow the modeling of both structural and temporal aspects of networks. By learning node representations that evolve over time, temporal graph models can capture how spammers interact differently from legitimate users across multiple time windows. For instance, while legitimate users typically maintain consistent posting behavior, spammers may show bursts of activity, sudden increases in connections, or coordinated campaigns that evolve rapidly.

In this paper, we propose a Dynamic Spammer Detection framework that combines the strengths of PCA, CNN, and Temporal Graph Embeddings (TGEs). PCA ensures efficient dimensionality reduction and noise filtering, CNNs capture local textual and content-based patterns, and TGEs model the evolving relational dynamics of users in the social graph. Together, this hybrid model offers a robust and adaptive approach to spammer detection in Twitter.

The contributions of this paper can be summarized as follows:

1. Hybrid architecture: We introduce a novel framework that integrates PCA-based dimensionality reduction, CNN-based content analysis, and temporal graph embeddings for dynamic spammer detection.
2. Temporal modeling of user behavior: Unlike static detection models, our approach explicitly captures the evolutionary dynamics of user interactions and posting patterns over time.
3. Comprehensive evaluation: We validate our model on a real-world Twitter dataset, comparing it with classical baselines and hybrid models, demonstrating significant improvements in accuracy, precision, recall, and F1-score.
4. Practical relevance: By modeling both content and temporal interaction patterns, the proposed method provides a more adaptive solution to the arms race between spammers and detection systems.

The remainder of this paper is organized as follows. Section 2 reviews related work in spam detection, including machine learning, deep learning, and graph-based approaches. Section 3 presents the proposed PCA–CNN–TGE model in detail. Section 4 describes the experimental setup and reports results. Section 5 discusses the findings, while Section 6 concludes the paper and highlights future directions.

2. Related Work

Spam detection in online social networks has been studied extensively over the past two decades, reflecting its importance in cybersecurity and information integrity. Various approaches have been proposed, ranging from traditional machine learning methods relying on handcrafted features to more sophisticated deep learning and graph-based models. In this section, we review the major categories of work and identify the gap that motivates our proposed model.

With classical machine learning approaches, early spam detection research was dominated by rule-based systems and feature-engineered machine learning models. These methods relied on manually selected indicators such as the presence of URLs, posting frequency, number of hashtags, follower–following ratios, or account age. Classifiers such as Support Vector Machines (SVMs), Naïve Bayes, Decision Trees, and Random Forests were applied to these features for supervised classification.

For instance, Gharge and Chavan [1] proposed a pipeline consisting of tweet collection, spam labeling, feature extraction, and classification using SVM. Their results showed accuracy levels between 95–97%, highlighting the effectiveness of carefully crafted features. Similarly, Fazil and Abulaish [2] demonstrated that community-based features (e.g., clustering coefficients, user connectivity patterns) combined with interaction-based features (e.g., frequency of replies or mentions) can significantly improve spammer detection performance when used with classical classifiers.

However, while these methods achieved good performance on static datasets, they suffered from several limitations. First, handcrafted features are brittle: once spammers change their strategies, these features lose discriminative power. Second, manual feature design is time-consuming and lacks scalability across platforms. Third, classical models often treat observations as independent samples, ignoring the sequential or relational nature of social interactions. These shortcomings paved the way for deep learning models, which automatically extract complex features from raw data.

With deep learning approaches, the rise of deep learning brought significant improvements to text classification, sentiment analysis, and anomaly detection, making it an attractive solution for spam detection in social networks. Convolutional Neural Networks (CNNs), initially popularized in computer vision, were successfully adapted for text data due to their ability to capture local n-gram patterns. Jain et al. [3] combined CNNs with Long Short-Term Memory networks (LSTMs) to leverage both local and sequential features in short-text spam detection tasks. Their model introduced a semantic embedding layer enriched by WordNet and ConceptNet, improving word representation in noisy short messages.

Other works explored hybrid deep learning models. Shahariar et al. [4] proposed combining CNN, LSTM, and Multi-Layer Perceptrons (MLPs) in a multi-stage architecture for spam review detection. Their experiments demonstrated that deep learning models consistently outperformed traditional classifiers such as Naïve Bayes, k-Nearest Neighbors, and SVM. Similarly, Alom et al. [5] developed a deep learning framework specifically for Twitter spam detection, employing word embeddings and stacked convolutional layers, showing superior accuracy over handcrafted approaches.

To reduce dimensionality and mitigate overfitting, dimensionality reduction techniques such as Principal Component Analysis (PCA) have also been integrated with deep models. The PCA–CNN hybrid approach [6] demonstrated that projecting high-dimensional word embeddings into a reduced feature space prior to convolution not only decreased training time but also improved classification performance by filtering noise. This confirmed that hybrid architectures combining statistical and deep learning techniques can be particularly effective in noisy environments like Twitter.

Nevertheless, most existing deep learning approaches share a common limitation: they treat the spam detection task as a static classification problem. Tweets and accounts are analyzed as isolated samples, ignoring how user behavior and relationships evolve over time. Spammers, however, adapt

dynamically, which static CNN or LSTM architectures fail to capture effectively. This shortcoming has motivated the exploration of graph-based and temporal models.

With graph-based and temporal approaches, social networks are naturally represented as graphs, where users are nodes and interactions (mentions, replies, retweets, follows) form edges. This has motivated the use of Graph Neural Networks (GNNs) for tasks such as spam detection, fake news classification, and bot identification. Kipf and Welling [7] introduced Graph Convolutional Networks (GCNs), while Hamilton et al. [8] proposed GraphSAGE, both demonstrating that relational context complements content-based features.

However, classical GNNs assume static graphs, whereas real-world networks evolve continuously. To address this, Temporal Graph Embedding (TGE) and Temporal Graph Neural Network (TGNN) models were proposed. Grover and Leskovec [9] introduced node2vec, and Perozzi et al. [10] proposed DeepWalk, which inspired temporal extensions. More recently, Rossi et al. developed TGAT [11] and TGN [12], enabling continuous-time embeddings that adapt as new interactions occur.

These temporal methods are highly relevant for spammer detection, since spammers display bursty or irregular dynamics unlike legitimate users. By combining PCA–CNN content modeling with TGEs, our framework bridges the gap between static deep learning and dynamic graph analysis.

3. Proposed Model

In this section, we present the proposed dynamic spammer detection framework, which integrates Principal Component Analysis (PCA) for dimensionality reduction, Convolutional Neural Networks (CNNs) for content-based representation, and Temporal Graph Embeddings (TGEs) for modeling the evolving relational behavior of users in social networks.

Due to the page limit of this template, we will only focus on the most important points in the remainder of the paper. The framework consists of five main stages:

1. Preprocessing and feature construction: Raw tweets and user metadata are preprocessed, tokenized, and converted into embeddings (e.g., Word2Vec). Social interactions (mentions, replies, retweets) are also logged to construct temporal graphs.
2. Dimensionality reduction (PCA): Word embeddings are transformed into compact, high-order feature vectors to eliminate noise and reduce computational overhead [6].
3. Content modeling (CNN): Reduced feature vectors are processed by a CNN module to capture discriminative textual patterns that distinguish spam content from legitimate communication [6].
4. Temporal relational modeling (TGE): User–user interaction graphs are embedded into a dynamic feature space using temporal graph embedding techniques [11], [12], capturing evolving behavioral patterns.
5. Fusion and classification: CNN-based content features and TGE-based relational features are concatenated and passed through fully connected layers to output a final classification (spammer vs. legitimate).

3.1 Feature Extraction via PCA

Given a set of preprocessed tweets represented by embedding matrices

$$X = \{x_1, x_2, \dots, x_n\}, \qquad x_i \in R^d \qquad (1)$$

where *d* is the embedding dimension (e.g., d=300), PCA is applied to reduce redundancy.

The covariance matrix is defined as

$$C = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)\,(x_i - \mu)^T \qquad (2)$$

where μ is the mean vector. Eigen decomposition yields

$$Cv_j = \lambda_j v_j \qquad (3)$$

with eigenvalues λ_j ranked in descending order. By selecting the top k eigenvectors, we project embeddings into a lower-dimensional subspace:

$$z_i = V_k^T\,(x_i - \mu), \qquad z_i \in R^d \qquad (4)$$

where V_k is the matrix of top k eigenvectors. This step filters noise while preserving most of the discriminative variance.

3.2 Content Modeling with CNN

The reduced feature vectors z_i serve as inputs to the CNN module. The CNN consists of:

- Convolutional layer: Applies filters $W \in R^{h\times k}$ across embeddings to capture local n-gram patterns.

$$C_j = f\left(W.z_{j:j+h-1} + b\right) \qquad (5)$$

  where $f(\cdot)$ is a ReLU activation.
- Pooling layer: Aggregates local features via max-pooling:

$$\hat{c} = \max\{c_1, c_2, \ldots, c_m\} \qquad (6)$$

- Fully connected layer: Produces high-level representation of text-based features.

This module captures textual cues (spam keywords, abnormal phrasing, repetitive content) associated with spammers.

3.3 Temporal Graph Embeddings

While CNN captures textual features, spammers also reveal themselves through interaction dynamics in the social graph. To capture these evolving behaviors, we employ the Temporal Graph Attention Network (TGAT) [11], a state-of-the-art framework for learning embeddings in continuous-time dynamic graphs.

Let the social network at time t be represented as a temporal graph:

$$G_t = (V, E_t, \tau) \qquad (7)$$

where V is the set of users, E_t is the set of edges (e.g., mentions, retweets, replies) observed up to time t, and $\tau(e)$ assigns a timestamp to each interaction $e \in E_t$.

Each user $v \in V$ is associated with a dynamic embedding $h_v^{(t)} \in \mathrm{R}^{\mathrm{d}}$, which evolves over time as new interactions occur. TGAT updates embeddings using two mechanisms:

- Time Encoding

Continuous time is encoded into a vector representation $\Phi(\Delta t)$ using sinusoidal basis functions:

$$\Phi(\Delta t) = [cos(w_1 \Delta t), sin(w_1 \Delta t), \dots, cos(w_k \Delta t), sin(w_k \Delta t)] \quad (8)$$

where $\Delta t = t - \tau(e)$ is the elapsed time since the last interaction, and ω_i are learnable frequencies. This encoding enables the network to incorporate fine-grained temporal information.

- Temporal Attention Aggregation

  For each node *v*, TGAT aggregates features from its temporal neighborhood *N(v,t)* using attention:

$$h_v^{(t)} = \sigma\left(\sum_{u \in N(v,t)} \alpha_{vu}^{(t)} . W\left[h_u^{(\tau(e))} \,||\, \Phi(\Delta t)\right]\right) \quad (9)$$

  where *W* is a learnable weight matrix, || denotes concatenation, σ is a non-linear activation (e.g., ReLU), and $\alpha_{vu}^{(t)}$ are attention coefficients computed as:

$$\alpha_{vu}^{(t)} = \frac{\exp(LeakyReLU(\alpha^T [Wh_v] \,||\, Wh_u \,||\, \Phi(\Delta t))}{\sum_{k \epsilon N(v,t)} exp(LeakyReLU(\alpha^T [Wh_v] \,||\, Wh_k \,||\, \Phi(\Delta t))} \quad (10)$$

  This formulation ensures that recent and relevant neighbors receive higher weights during aggregation, reflecting the intuition that spammers' latest actions are often the most suspicious.

  The output is a temporally aware embedding vector $h_v^{(t)}$ that captures both the structural context and the temporal evolution of a user's behavior.

  By integrating TGAT embeddings with PCA–CNN content features, our model can simultaneously learn from what a user posts (content) and how their behavior changes over time (temporal interactions). This dual perspective makes the classifier robust to evolving spammer tactics.

4. Experiments and Results

In this section, we evaluate the proposed PCA–CNN–TGAT framework on a benchmark dataset and compare its performance with state-of-the-art baselines. We conducted the experiments on the Cresci-2017 Dataset[3]. This dataset provides multiple types of Twitter accounts, including genuine users, spambots, and social spambots that mimic real users' behavior. The dataset is preprocessed to remove inactive users, non-English tweets, and duplicated accounts. We split the data into 70% training, 15% validation, and 15% test sets.

4.1 Experimental Setup

- Text preprocessing: Tweets were lowercased, tokenized, and stop words removed. Each token was mapped into a 300-dimensional Word2Vec [4]embedding.
- Dimensionality reduction (PCA): Embedding vectors were reduced from d=300 to k=100 dimensions, preserving ~95% variance.
- CNN configuration: One convolutional layer with 128 filters of sizes [3,4,5], followed by max-pooling, ReLU activation, and dropout (rate = 0.5).

[3] https://botometer.osome.iu.edu/bot-repository/datasets.html

[4] https://arxiv.org/abs/1301.3781

- TGAT configuration: Node embeddings dimension = 128; time encoding dimension = 64; 2 attention heads; memory update = GRU cell. We should mention here that GRU is chosen because it's lighter than LSTM but still powerful for sequence updates.
- Fusion: CNN and TGAT outputs were concatenated into a 256-dimensional feature vector, fed into a 2-layer fully connected network with softmax output.
- Training: learning rate = 0.001, batch size = 128, epochs = 30. Training was conducted on an NVIDIA GPU with PyTorch Geometric Temporal library.

4.2 Results and evaluation

To evaluate the proposed PCA–CNN–TGAT model, we compared it against three representative baselines:

- SVM: a traditional machine learning approach using handcrafted features.
- CNN: a deep learning baseline that captures textual content features without dimensionality reduction.
- PCA–CNN: our previous baseline model, where PCA reduces noise and redundancy before CNN-based classification.

Table 1 reports the performance on the Cresci-2017 dataset.

Table1. Performance comparison against representative baselines

Model	Accuracy	Precision	Recall	F1-score	AUC-ROC
SVM	0.87	0.84	0.82	0.83	0.89
CNN	0.91	0.88	0.87	0.87	0.92
PCA–CNN	0.93	0.90	0.89	0.90	0.94
PCA–CNN–TGAT	0.96	0.94	0.93	0.94	0.97

The results show that:

1. SVM lags behind, confirming the weakness of handcrafted features for evolving spammer strategies.
2. CNN improves performance by automatically extracting content features.
3. PCA–CNN further enhances CNN by filtering redundant features, yielding a stronger baseline.
4. Our PCA–CNN–TGAT achieves the best performance, highlighting the effectiveness of incorporating temporal graph embeddings to capture evolving user behaviors.

To evaluate the contribution of the temporal graph embedding (TGAT) component, we performed an ablation experiment comparing our previous PCA–CNN baseline with the proposed PCA–CNN–TGAT model.

- PCA–CNN (without TGAT): This configuration relies only on textual features processed through PCA and CNN. It achieved an F1-score of 0.90, showing strong content-based discrimination.
- PCA–CNN–TGAT (full model): By incorporating temporal graph embeddings, the model achieved an F1-score of 0.94, a clear improvement over the PCA–CNN baseline.

The improvement in F1-score from 0.90 (PCA–CNN) to 0.94 (PCA–CNN–TGAT) confirms that temporal dynamics play a crucial role in distinguishing spammers from legitimate users. This indicates that while PCA–CNN is effective at capturing static content features, the addition of TGAT allows the model to exploit temporal behavioral patterns, leading to higher detection accuracy and robustness against evolving spammer tactics.

5. Conclusion and Future Work

In this paper, we proposed a dynamic spammer detection framework that combines dimensionality reduction (PCA), content modeling (CNN), and temporal graph embeddings (TGAT). Unlike traditional machine learning or static deep learning approaches, our model captures not only the textual features of user-generated content but also the evolving dynamics of user interactions in social networks.

Experiments on the Cresci-2017 Twitter dataset demonstrated that the proposed PCA–CNN–TGAT model significantly outperforms both classical baselines (SVM) and deep learning approaches (CNN, PCA–CNN). In particular, the integration of temporal embeddings improved the F1-score from 0.90 (PCA–CNN) to 0.94, highlighting the importance of modeling temporal behavioral patterns for robust spammer detection.

For future work, we plan to extend this framework in two main directions. First, we aim to develop real-time detection mechanisms, enabling the system to flag spammers as interactions occur rather than after large batches of data are collected. Second, we intend to evaluate the model on multi-platform datasets (e.g., Facebook and Instagram) to assess its robustness and adaptability across different social network environments.

References

[1] S. Gharge and M. Chavan, "An integrated approach for malicious tweets detection using NLP," in Proc. Int. Conf. Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, pp. 435–438, 2017. doi: 10.1109/ICICCT.2017.7975235.

[2] M. Fazil and M. Abulaish, "A hybrid approach for detecting automated spammers in Twitter," IEEE Trans. Inf. Forensics Security, vol. 13, no. 11, pp. 2707–2719, Nov. 2018. doi: 10.1109/TIFS.2018.2825958.

[3] G. Jain, M. Sharma, and B. Agarwal, "Spam detection in social media using convolutional and long short-term memory neural network," Ann. Math. Artif. Intell., vol. 85, pp. 21–44, Jan. 2019. doi: 10.1007/s10472-018-9612-z.

[4] G.M. Shahariar, S. Biswas, F. Omar, F.M. Shah, and S.B. Hassan, "Spam review detection using deep learning," in Proc. IEEE 10th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON), Vancouver, BC, Canada, pp. 27–33, 2019. doi: 10.1109/IEMCON.2019.8936148.

[5] Z. Alom, B. Carminati, and E. Ferrari, "A deep learning model for Twitter spam detection," Online Social Netw. Media, vol. 18, 100079, Feb. 2020. doi: 10.1016/j.osnem.2020.100079.

[6] H. Loucif, "A hybrid deep learning approach for spam detection in Twitter," Ingénierie des Systèmes d'Information, vol. 29, no. 1, pp. 117–123, Feb. 2024. doi: 10.18280/isi.290113.

[7] T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907, 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1609.02907

[8] W. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, pp. 1024–1034, 2017.

[9] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD)*, San Francisco, CA, USA, pp. 855–864, 2016. doi: 10.1145/2939672.2939754.

[10] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. 20th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD)*, New York, NY, USA, pp. 701–710, 2014. doi: 10.1145/2623330.2623732.

[11] E. Rossi, B. Chamberlain, F. Frasca, D. Eynard, F. Monti, and M. Bronstein, "Temporal graph attention networks for deep learning on dynamic graphs," in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, Long Beach, CA, USA, pp. 4937–4947, 2020.

[12] E. Rossi, B. Chamberlain, M. Bronstein, and F. Monti, "Temporal graph networks for deep learning on dynamic graphs," *arXiv preprint* arXiv:2006.10637, 2020. [Online]. Available: https://doi.org/10.48550/arXiv.2006.10637