IOHunter: Graph Foundation Model to Uncover Online Information Operations

Marco Minici[1,2], **Luca Luceri**[3,4], **Francesco Fabbri**[5], **Emilio Ferrara**[3,4]
[1]ICAR-CNR, Rende, Italy
[2]University of Pisa, Pisa, Italy
[3]Information Sciences Institute, University of Southern California, Marina Del Rey, CA, USA
[4]Thomas Lord Department of Computer Science, University of Southern California, Los Angeles, CA, USA
[5]Spotify, Barcelona, Spain
marco.minici@icar.cnr.it, lluceri@isi.edu, francescof@spotify.com, emiliofe@usc.edu

Abstract

Social media platforms have become vital spaces for public discourse, serving as modern agorás where a wide range of voices influence societal narratives. However, their open nature also makes them vulnerable to exploitation by malicious actors, including state-sponsored entities, who can conduct information operations (IOs) to manipulate public opinion. The spread of misinformation, false news, and misleading claims threatens democratic processes and societal cohesion, making it crucial to develop methods for the timely detection of inauthentic activity to protect the integrity of online discourse. In this work, we introduce a methodology designed to identify users orchestrating information operations, a.k.a. IO drivers, across various influence campaigns. Our framework, named IOHunter, leverages the combined strengths of Language Models and Graph Neural Networks to improve generalization in supervised, scarcely-supervised, and cross-IO contexts. Our approach achieves state-of-the-art performance across multiple sets of IOs originating from six countries, significantly surpassing existing approaches. This research marks a step toward developing Graph Foundation Models specifically tailored for the task of IO detection on social media platforms.

Code — https://github.com/mminici/SocGFM
Datasets — https://zenodo.org/records/13357621

Introduction

Online social media platforms have become essential for fostering public discourse, where users engage in debates on critical political and social issues. The integrity of these online spaces is paramount, given their significant role in shaping public opinion and influencing societal outcomes, such as elections or public health interventions (Starbird 2019; Ferrara 2015; Nogara et al. 2022). However, these platforms are increasingly vulnerable to state-sponsored Information Operations (IOs), which seek to manipulate narratives, spread disinformation, and foster division through the promotion of hate speech and other harmful content (Badawy, Ferrara, and Lerman 2018; Zannettou et al. 2019; Suresh et al. 2024; Minici et al. 2024). The proliferation of such campaigns poses a significant threat to democratic processes, highlighting the urgent need for robust methods to detect and mitigate these operations (World Economic Forum 2024).

The development of machine learning techniques for IO detection is a rapidly growing area of research. Recent studies, such as Luceri et al. 2024, have demonstrated the potential to leverage graph machine learning techniques for this purpose. Specifically, they leverage node2vec embeddings of similarity networks constructed from behavioral traces, such as co-sharing patterns, to detect coordinated users driving IOs, namely *IO drivers*. Their findings highlight the potential of using topological structures based on similarity patterns, combined with graph machine learning techniques, for detecting IOs. However, they did not explore whether recent advancements in Graph Neural Networks (GNNs) could further improve performance. GNN-based approaches not only provide a more powerful framework for modeling online user behavior but also offer inductive capabilities, enabling generalization to nodes not seen during training. This flexibility is particularly crucial for deploying auditing tools in dynamic environments, where threats can arise from new users or, even more critically, from IOs originating in different geopolitical contexts.

Generalizing across different IOs—referred to here as *cross-IO detection*—is inherently difficult, as distinct IOs often employ different coordination strategies and may operate in different languages (Luceri et al. 2024).

Contribution of this work. In this paper, we propose IOHunter, an architecture for IO detection that combines the message-passing paradigm of GNNs with multi-modal information derived from both network structure and textual content. Unlike existing approaches that are either based on graph structure or textual content, IOHunter builds on the emerging concept of Graph Foundation Models (GFMs). Traditional GNNs are typically trained from scratch on specific tasks and datasets, limiting their ability to generalize across different domains. In contrast, IOHunter integrates GNNs with embeddings extracted from Language Models to create a GFM capable of leveraging large-scale, diverse graph data with the goal of rapidly adapting to new tasks or datasets. Our approach is thoroughly detailed in the Methodology section. We evaluate IOHunter on six datasets from Twitter, each representing an IO originating from distinct geopolitical contexts: UAE, Cuba, Russia, Venezuela, Iran, and China. IOHunter achieves improvements of up to

+20% in Macro-F1 compared to the state-of-the-art in IO detection. Additionally, we demonstrate the robustness of `IOHunter` in scenarios with limited data availability and its effectiveness in cross-IO detection tasks when pretrained and fine-tuned with minimal labeled data.

Related Work

Machine Learning-Based IO Detection

Research in IO detection has extensively analyzed individual account activities to detect participation in influence campaigns, particularly focusing on bots (software-controlled accounts) and trolls (state-backed human operators) (Mazza et al. 2022; Ferrara 2023). Bot detection has been a focal point, with various solutions utilizing machine learning strategies to *(i)* identify bot characteristics, such as posting frequency, content patterns, and network behavior (Yang et al. 2019; Chen and Subramanian 2018; Cresci et al. 2016), and/or *(ii)* distinguish patterns of bot behavior from organic human behavior (Pozzana and Ferrara 2020). Notably, the Botometer tool (Yang et al. 2019; Yang, Ferrara, and Menczer 2022) has played a significant role in scaling bot activity research on Twitter, enabling studies focused on the identification of bot-driven influence campaigns (Shao et al. 2018; Stella, Ferrara, and De Domenico 2018; Deb et al. 2019; Grinberg et al. 2019; Luceri et al. 2019).

However, recent studies have emphasized that IO coordination extends beyond automated bots, highlighting the role of human-operated trolls in these operations (Nizzoli et al. 2021; Hristakieva et al. 2022). Research on state-sponsored trolls has been categorized into three primary detection methods: content-based, behavioral-based, and sequence-based approaches. Content-based methods analyze the linguistic features of posts to identify deceptive or coordinated messaging (Alizadeh et al. 2020; Luceri, Boniardi, and Ferrara 2024; Im et al. 2020). Behavioral-based approaches focus on user activity patterns, such as posting activity and interaction signals, to detect coordinated inauthentic behavior (Luceri, Giordano, and Ferrara 2020; Kong et al. 2023; Sharma et al. 2021). Sequence-based techniques, on the other hand, model the temporal sequence of actions to uncover orchestrated activities over time (Nwala, Flammini, and Menczer 2023; Ezzeddine et al. 2023).

Network-Based IO Detection

In addition to machine learning-based methods, a significant body of research has focused on detecting IOs through network-based approaches. These methods aim to uncover tactics of online coordination by identifying unexpected or exceptional similarities in the actions of multiple users (Pacheco, Flammini, and Menczer 2020; Pacheco et al. 2021; Nizzoli et al. 2021; Mannocci et al. 2024; Magelinski, Ng, and Carley 2022; Luceri et al. 2024). The underlying assumption is that connections between highly similar users—such as those who share the same content, use similar hashtags, or post at synchronized times—can reveal coordinated clusters likely engaged in IOs.

Network-based detection typically involves constructing networks that represent user similarities using edge weights, where higher weights indicate stronger behavioral correlations. By exploiting network properties to filter out organic users, researchers identify clusters of users exhibiting collective similarity and potentially driving IOs (Pacheco et al. 2021; Luceri et al. 2024). This approach has proven effective in revealing coordinated activities, providing valuable insights into the structure and scale of influence campaigns.

Graph Foundation Model

The challenge of training models capable of generalizing across diverse graph domains and tasks has recently attracted significant attention (Mao et al. 2024). Earlier work in this area has focused on self-supervised approaches to enable rapid adaptation to downstream tasks on the same graph (Lu et al. 2021; De Nadai et al. 2024) or to generalize across graphs from different domains (Qiu et al. 2020; Jiang et al. 2021). However, these methods do not address the challenge of integrating multi-modal information.

Recent advancements have explored the use of LMs to generate transferable features in heterogeneous graph settings, such as personalization (Damianou et al. 2024) and e-commerce applications (Xie et al. 2023). Others, like PRODIGY (Huang et al. 2024) and OFA (Liu et al. 2024), focus on adapting graph tasks to leverage the generalization capabilities of LLMs for in-context learning tasks.

In contrast, our approach specifically targets the problem of IO detection across three distinct learning regimes. Within this context, we demonstrate that integrating multi-modal signals combined with massive pre-training—while keeping the LM weights frozen—is an effective strategy to achieve a GFM tailored to the IO detection task.

Problem Definition

We are given an undirected graph $G = (V, E)$, where $V = \{v_1, ..., v_n\}$ represents the set of nodes, and $E \subset V \times V$ denotes the set of edges. In this context, G models the relationships between social media users, with an edge $\ell(v_1, v_2) \in E$ existing if two users v_1 and v_2 are considered similar. We will also refer to G as the similarity network[1]. For each user $v_i \in V$, we have access to a set of content C_i (e.g., texts, images) that v_i has shared on the social network. Additionally, each user v_i is associated with a label $y_i \in \{0, 1\}$, where 1 indicates an *IO driver* and 0 represents a legitimate user.

Our objective is to learn two functions: a multi-modal projection $p_\psi : V \to \mathbb{R}^d$ and a probabilistic node classifier $f_\theta : \mathbb{R}^d \to [0, 1]$. The multi-modal projection p_ψ maps each node v_i to a point $z_i = p_\psi(v_i \mid G, C_i)$ in a d-dimensional latent space, utilizing both the contextual information from G and the content C_i shared by the user. For simplicity, we will refer to this embedding as $p_\psi(v_i)$.

The node classifier f_θ takes the low-dimensional representation z_i as input and outputs a score $s_i = f_\theta(z_i)$, indicating the likelihood that v_i is an *IO driver*. Given that our task is a binary classification problem, we optimize the complete set of learnable parameters $\Theta = \{\psi, \theta\}$ by minimizing

[1]We use network and graph interchangeably.

the Binary Cross-Entropy loss for each node v_i as follows:

$$\mathcal{L}(\Theta, v_i) = -\left[y_i \log(s_i) + (1 - y_i) \log(1 - s_i)\right]. \quad (1)$$

Methodology

Our objective is to detect *IO drivers* by integrating two sources derived from the behavioral traces of social media users: *(i)* the textual content they share and *(ii)* the similitude of their sharing activities as captured by similarity networks. Our approach, illustrated in Fig. 1, begins by extracting textual embeddings from user-shared posts and graph embeddings from the *Fused Similarity Network* (see below). These two data modalities are then blended using a cross-attention module, allowing the model to learn interactions between the content and network contexts. The resulting multi-modal embeddings are subsequently input into a GNN, which leverages this enriched representation to accurately predict user categories.

Fused Similarity Network. The process of constructing a similarity graph generally follows a consistent approach in the literature (Pacheco, Flammini, and Menczer 2020; Pacheco et al. 2021; Luceri et al. 2024). We consider different similarities — including the sharing of identical links (*co-URL*), hashtags (*co-hashtag*), or content (*text similarity*), the re-sharing of the same tweets (*co-retweet*), and automation-driven actions such as rapid retweeting (*fast-retweet*) — to build five distinct similarity networks. We construct a bipartite graph between users and entities, where the entities correspond to the specific behavioral trace being analyzed (e.g., for the Co-URL trace, the entities are the URLs). The Fast Retweet bipartite network is constructed similarly to the co-retweet bipartite network, but it excludes connections where the retweet takes more than 10 seconds. In this bipartite network, users are linked to entities based on their sharing activities, with weights assigned using TF-IDF to account for the popularity of each entity. Consequently, each user is represented as a TF-IDF vector of the shared entities. This bipartite graph is then transformed into a similarity network, where users are connected based on the similarity of their behavioral traces.

Building on (Luceri et al. 2024), we combine the five similarity networks by linking two nodes in a *Fused Similarity Network* if they are connected in any of the individual similarity networks.

Content Embedding. For each user v_i, we extract a low-dimensional embedding from its set of shared content C_i. Since the main source of information in our experiments is textual content, we use a Sentence Transformer, SBert (Reimers and Gurevych 2019). We refer to $c_i \in \mathbb{R}^{d_c}$ as the content embedding of v_i, where d_c is output dimensionality of SBert. If $|C_i| > 1$ then c_i will be the average of the embeddings of each single content in C_i.

Contextual Embedding. To consider the contextual information provided by the graph G, we abide by the best practice (Cui et al. 2022), and divide degree values into d_g buckets, then map the degree value distributed in each bucket range into one class, and finally construct a unique one-hot vector for each class. Hence, we encode the structural information of each node v_i to a one-hot vector $g_i \in \{0, 1\}^{d_g}$.

Multi-Modal Blend. Given the content and contextual embeddings c_i and g_i, we now need to merge them in a unique projection z_i that can be used by a node classifier to detect if the node v_i is an *IO driver*. We propose to use a cross-attention mechanism, loosely inspired by (Abavisani et al. 2020). The main advantage of this approach is to filter out any irrelevant information that might come from the other modality, thus allowing to learn how to best combine content and contextual information.

First, we remap both c_i and g_i to a $d-$dimensional space using a fully-connected layer and a $ReLU$ non-linear activation as follows:

$$\tilde{c}_i = ReLU(W_c^T c_i + b_c), \quad (2)$$

$$\tilde{g}_i = ReLU(W_g^T g_i + b_g). \quad (3)$$

We also compute the cross-attention coefficients α_c, α_g for each modality:

$$\alpha_c = ReLU(W_c'^T g_i + b_c'), \quad (4)$$

$$\alpha_g = ReLU(W_g'^T c_i + b_g'). \quad (5)$$

The node representation z_i is obtained by an element-wise multiplication of cross-attention coefficients and content/-contextual embeddings, followed by a concatenation operation:

$$z_i = \alpha_c \odot \tilde{c}_i \parallel \alpha_g \odot \tilde{g}_i. \quad (6)$$

The final node representation z_i is further refined by two fully-connected layers $W_z^{(1)}, b_z^{(1)}, W_z^{(2)}, b_z^{(2)}$, each one followed by a non-linear ReLU function.

We will refer to all these steps using the function $z_i = p_\psi(v_i)$, following the notations used in the *Problem Definition* section.

GNN Model. The derived multi-modal node embedding $z_i = p_\psi(v_i)$ is given as input to a function f_θ that outputs the probability of v_i to be an *IO driver*. We use a generic GNN model (Wu et al. 2020) (with a final logit head) to model f_θ, given the state-of-the-art performances of these models on graph-related tasks. Specifically, we leverage GCN (Kipf and Welling 2022) and Sage (Hamilton, Ying, and Leskovec 2017) models.

The overall architecture is graphically described in Figure 1, while the the end-to-end learning procedure is shown in Algorithm 1.

Experiments

This experimental section examines whether our `IOHunter` can detect IOs on social media networks. We address the following research questions investigating the applicability of our proposal in different, realistic data regimes:

- **RQ1:** What performance does `IOHunter` achieve in a supervised IO detection task compared to state-of-the-art methods?
- **RQ2:** In a more realistic scenario with limited labeled data, can `IOHunter` maintain strong predictive performance and outperform the best state-of-the-art method in this data regime?

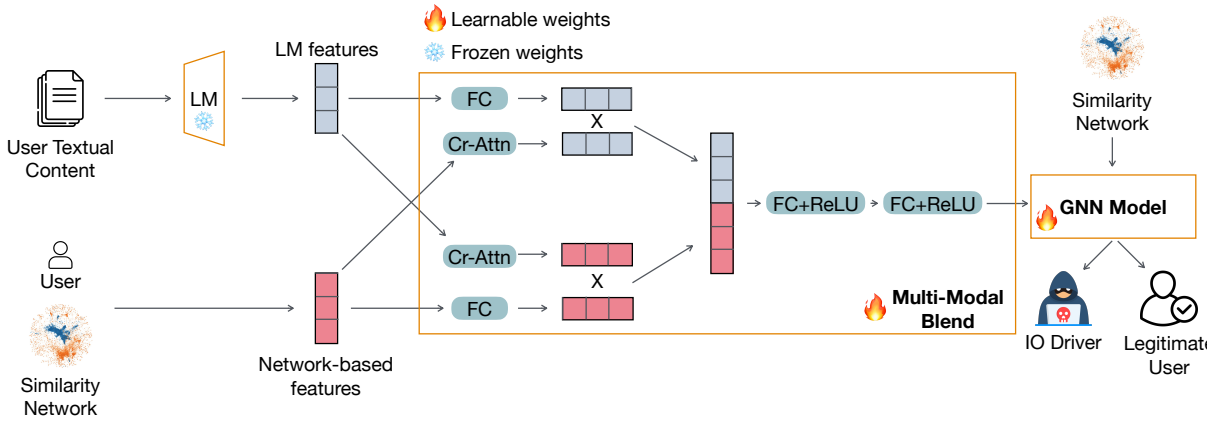


Figure 1: Illustration summarizing IOHunter. We feed the user posts to a multi-lingual SBert and then average all post embeddings to obtain a unique textual representation. SBert is frozen and not optimized. We also extract a degree-based embedding from the fused similarity network. Both modality embeddings are blended through a cross-attention layer and a couple of fully connected layers after concatenation. The obtained multi-modal representation is then fed to a GNN model that determines whether the user is an IO Driver or a legitimate user. Both the multi-modal embedding module and the GNN model are optimized during the training phase.

Algorithm 1: Learning Procedure of IOHunter

Input: Graph $G = (V, E)$,
User Content $C = \{C_i \mid \forall v_i \in V\}$
Result: Model parameters Θ

Compute c_i and g_i for all users $v_i \in V$
Initialize $\Theta^{(0)}$
while *training has not converged* **do**
  Sample $V_{\text{batch}} \subseteq V$
  $l_{\text{tot}} = 0$
  forall $v_i \in V_{batch}$ **do**
    l_{tot} += $\mathcal{L}(\Theta^{(t)}, v_i)$ // eq. 1
  $\Theta^{(t+1)} \leftarrow$ BackPropagate(l_{tot}) // Update model

- **RQ3:** Does a pre-training on a massive dataset enable IOHunter to generalize effectively across unseen IOs?

Experimental Settings

Datasets. We conduct our experiments on 6 datasets from (Seckin et al. 2024), including IO activities in countries such as UAE, Cuba, Russia, Venezuela, Iran, and China. In accordance with previous studies (Luceri et al. 2024), we select these state-sponsored campaigns for their extensive scale, evident from their size in Table 1. Each country can include multiple campaigns, mirroring real-world situations where both campaign-based and organic conversations from a single country might intersect.

The datasets exhibit different properties in terms of number of nodes, density, edge homophily and label imbalance. We use the class-insensitive edge homophily (Lim et al. 2021) to evaluate the degree of homophily in each graph. Notably, there is significant variation in homophily across the datasets, ranging from approximately 37% to 81%. The presence of heterophilic edges may impact the effectiveness of standard GNN models (Zheng et al. 2022).

Country	Nodes	Edges	Homophily	IO Prop.
UAE	9242	2118684	52.8%	35.7%
Cuba	19822	4737374	37.1%	2.3%
Russia	666	10381	53.0%	38.4%
Venezuela	4980	56700	77.7%	10.6%
Iran	12977	392938	81.0%	32.2%
China	22694	410979	41.1%	3.3%

Table 1: Dataset statistics for different countries. Homophily indicates the percentage of edges connecting nodes of the same class, and IO Prop. indicates the proportion of IO drivers.

Models. To evaluate IOHunter performances, we compare it with state-of-the-art methods for the task of IO detection. We select two methods introduced by Luceri et al. (2024). The first one, **NodePruning** categorizes a user as an IO driver depending on their eigenvector centrality in the Fused Similarity Network. If this centrality value exceeds a certain threshold, the user is labeled as an IO driver. The second is based on node2vec (**Node2vec+RF**), which extracts node embedding based on the local network topology. The node2vec representations of the Fused Similarity Network are then used as input features for a Random Forest classifier to detect whether a user is an IO driver or not. To extend the set of baselines, we also include two more architectures. The first one is a shallow multilayer perceptron (**MLP**) trained on content-based features (**SBert+MLP**). Each user is represented by the average of their top 5 most popular tweets' embedding. Those are extracted using a sentence transformer, SBert (Reimers and Gurevych 2019). The last type of archi-

Input Features	Model	Dataset						
		UAE	Cuba	Russia	Venezuela	Iran	China	**Average**
Graph	NodePruning	84.66± 0.63	57.92± 2.07	87.65± 1.95	95.05± 1.47	60.83± 1.09	63.66± 0.70	74.96
	node2vec+RF	96.97± 0.42	91.53± 1.11	83.43± 3.24	90.32± 2.14	80.50± 0.60	83.89± 1.06	87.77
Text	SBert	86.23± 0.54	90.43± 1.17	84.03± 0.91	92.74± 1.38	85.18± 0.90	75.85± 1.53	85.74
Graph+Text	GNN	98.53± 0.13	85.22± 17.9	90.04± 2.66	97.46± 0.91	95.12± 0.30	66.42± 21.1	88.80
	IOHunter	**98.76± 0.14**	**98.93± 0.38***	**92.28± 2.24***	**99.11± 0.32***	**96.61± 0.35***	**92.90± 1.00***	**96.43**

Table 2: Results in terms of Macro-F1 for various models across datasets on the task of IO driver detection. The table reports both the average and standard deviation across five different random seeds. The best results are highlighted in bold, while the second-best are underlined. A statistically significant improvement between the best and second-best results (i.e., $p < 0.05$) according to a two-sided t-test is indicated with a star.

tecture we also test is based on GNNs. We test two popular types, GCN and Sage, with either only structural features, only textual features, or a simple concatenation of structural and textual features. In this manuscript, we present only the best-performing approach among these, i.e., the one trained with concatenation of text and graph-based features.

Implementation details. We utilize the PyTorch Geometric (Fey and Lenssen 2019) library to implement all baselines (except Node Pruning) and IOHunter. Since IO detection is a binary classification task, we use Macro-F1 as an evaluation metric to address the label imbalance, which can be noted in Table 1. We follow a 60-20-20 random split strategy to build training, validation, and test sets. We report average and standard deviation across five different seeds to obtain reliable performance estimates. We optimize each model for 1000 epochs, evaluating the Macro-F1 on the validation set at the end of each epoch, with early stopping implemented after a certain number of epochs of no improvement. We perform a hyper-parameter search on the learning rate of the Adam optimizer in $\{10^{-2}, 10^{-3}\}$, on the early stopping number of epochs in $\{20, 25, 30\}$ and on the number of MLP layers in $\{2, 3, 4\}$ fed with SBert embeddings. We set the number of latent dimensions to 128 for node2vec, following the configuration of (Luceri et al. 2024) and set the same number of hidden neurons for all other methods. All models based on GNNs use 2 message-passing layers. All neural models use Dropout units with 20% percentage. We use a full-batch procedure in Algorithm 1, hence V_{batch} is always equal to V. Experiments are conducted on a DGX Server equipped with 4 NVIDIA Tesla V100 GPU (32GB) and CUDA Version 12.2.

Supervised Detection of IOs (RQ1)

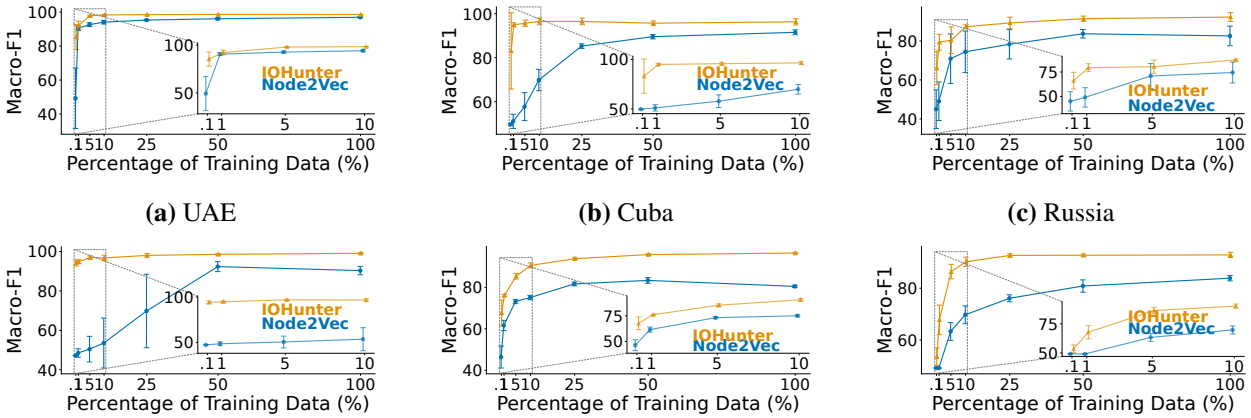
Table 2 shows the performances of IOHunter and compares it with other baselines and existing approaches. Our proposed method demonstrates a clear and consistent improvement over the state-of-the-art node2vec+RF across all six datasets, with an average percentage gain of 9.25% in terms of Macro-F1. The performance enhancement is particularly noteworthy, ranging from a gain of +1.8% on the UAE dataset to a +20% on the Iran dataset. This breadth of improvement across diverse datasets underscores the robustness and versatility of our approach, especially in scenarios where traditional methods like node2vec+RF may not fully capture the complexity of the data and diversity of IOs.

A critical factor in IOHunter success is the effective integration of both structural and textual features, which proves essential for achieving competitive results across all datasets. Relying exclusively on either structural or textual features, as evidenced by the inconsistent performance of the NodePruning and SBert models, falls short of delivering reliable outcomes across IOs. The need to combine both modalities is paramount, as neither feature set alone can encapsulate the full complexity required for effective user classification. The strength of our approach lies in its ability to seamlessly blend these modalities, which is a key driver behind its superior performance across diverse datasets.

Moreover, the inclusion of a cross-attention mechanism in our model significantly enhances performance compared to a straightforward multi-modal concatenation approach (i.e., GNN in Table 2), as evidenced by superior outcomes on 6 out of the 6 datasets. This improvement is not only reflected in higher absolute performance but also in greater stability. The cross-attention mechanism mitigates the variability often observed in models relying on simple concatenation, which can be sensitive to initial conditions such as random seed variations. By effectively capturing the interactions between different data modalities, the cross-attention mechanism enables a more reliable integration, ensuring consistent and robust results.

IO Detection under Data Scarcity (RQ2)

To evaluate the robustness and effectiveness of IOHunter, we conducted a series of experiments designed to test its performance under varying levels of data scarcity. We simulated different degrees of sparsification by downsampling the training set to create sparser versions. Specifically, we train our model using only the 0.1%, 1%, 5%, 10%, 25%, and 50% of the original training data. For each level of sparsification, we trained both IOHunter and node2vec+RF, which is the leading approach in the literature for detecting IOs (Luceri et al. 2024). This experimental design allows us to assess the robustness of IOHunter in scenarios where labeled data is extremely limited, a common challenge in



(a) UAE **(b)** Cuba **(c)** Russia

(d) Venezuela **(e)** Iran **(f)** China

Figure 2: Performance of IOHunter and Node2Vec+RF with different amount of training data sparsity.

real-world applications.

Figure 2 portrays the results of this analysis. IOHunter consistently outperforms the node2vec+RF across all levels of sparsification and across all six datasets. This consistent superiority across various data regimes highlights the robustness of our model and underscores its potential applicability in real-world scenarios where data availability can often be a limiting factor in detecting new, unseen IOs.

In particular, the performance gap between IOHunter and node2vec+RF becomes increasingly pronounced as the training data is reduced, especially in the datasets for Cuba, Russia, Venezuela, and China. For these countries, as the level of sparsification increases, IOHunter maintains a relatively stable performance while the compared model deteriorates significantly.

Notably, in the UAE, Cuba, and Venezuela datasets, even with as little as 0.1% of the training data, IOHunter is able to achieve a Macro-F1 score of approximately 80%. This is a remarkable result, especially given the extremely limited amount of training data. However, it is also worth noting that with such sparse data, there is more significant variability in the results across different random seeds.

Generalizable Cross-IO Detection (RQ3)

To evaluate the cross-IO generalization capability of IOHunter, we designed an experiment inspired by the principles underlying foundation models in computer vision (Zhai et al. 2022; Cherti et al. 2023) and natural language processing (Kaplan et al. 2020; Hernandez et al. 2021). The goal of this experiment is to determine whether pretraining the model on a diverse set of countries can enable it to generalize effectively to a new country for which no training data has been seen during pretraining. Also, we assess whether fine-tuning the pretrained model on a small fraction of data from the test country can further enhance its performance, particularly when labeled data is scarce.

In this experiment, we employ a leave-one-out strategy where the model is pretrained on data from all countries except one, which is reserved as the test country. We repeat this process for each of the six countries: Cuba, Russia, Venezuela, China, United Arab Emirates (UAE), and Iran. Following the pretraining phase, we evaluate the model's performance on the test country both in its pretrained state (referred to as the "Only PreTrain" strategy) and after fine-tuning it on just 0.1% of the training data from the test country (referred to as the "PreTrain & FineTune on 0.1%" strategy). The results are then compared against a baseline model that is trained only on 0.1% of the test country's training data without any pretraining.

Cross-country generalization experiments, presented in Table 3, highlight the significant advantages of our pretraining strategy. The "Only PreTrain" strategy, which involves pretraining on all countries except the test country, achieves remarkable performance. In four out of the six datasets, this strategy surpasses the performance of a model trained in a fully-supervised fashion on 0.1% of the test country's training data. This outcome demonstrates the model's strong ability to generalize across different datasets, effectively transferring knowledge from one domain to another. Furthermore, when we apply the "PreTrain & FineTune on 0.1%" strategy, the performance improves even further. Fine-tuning the pretrained model on a tiny percentage of the test country's data leads to an average improvement of more than 10% in Macro-F1 scores across all countries. This result underscores the power of combining massive pretraining with targeted fine-tuning, as it allows the model to adapt to the characteristics of the new domain with minimal supervision.

These findings suggest that pretraining on a diverse set of countries not only equips the model with a strong baseline capability for cross-country generalization but also that fine-tuning on a small fraction of data from the target country can yield substantial performance gains. This approach is particularly valuable in scenarios where labeled data is scarce or expensive to obtain.

Ours	Datasets						
	UAE	Cuba	Russia	Venezuela	Iran	China	Δ% No PreTraining
Train on 0.1%	85.33± 7.38	83.09± 17.3	66.13± 8.52	**94.16± 1.51**	67.65± 6.20	53.55± 3.33	----
Only PreTrain	83.93± 5.93	89.91± 5.35	79.77± 1.93	90.99± 1.07	72.78± 1.43	58.14± 5.89	+5.69%
PreTrain & FineTune on 0.1%	**88.97± 4.37**	**91.27± 3.12**	**85.09± 2.41**	92.10± 1.95	**73.75± 1.27**	**64.88± 1.78**	+10.25%

Table 3: Results of IOHunter in terms of Macro-F1 for the cross-country detection task. For each country c, IOHunter is pretrained on data from all other countries (i.e., Only PreTraining) and then tested on c's test set. IOHunter can be further fine-tuned on a tiny percentage (0.1%) of c training set. We compare the model trained on the 0.1% of c's training set.

Model	UAE	Cuba	Russia	Venezuela	Iran	China	**Average**
IOHunter w/o Graph	98.46	96.16	89.80	86.86	91.73	57.22	86.76
IOHunter w/o Text	82.54	86.86	88.04	98.34	84.35	91.05	88.53
IOHunter w/o CrossAttn	98.53	85.22	90.04	97.46	95.12	66.42	88.80
IOHunter	**98.76**	**98.93**	**92.28**	**99.11**	**96.61**	**92.90**	**96.43**

Table 4: Results of the ablation study on the proposed architecture, i.e. IOHunter. Metrics represent Macro-F1 on the test set. The full configuration of IOHunter consistently achieves the best results, highlighted in bold.

Ablation Study

We present the results of experiments designed to evaluate the importance of three key components of IOHunter: the cross-attention mechanism for integrating the two modalities, the graph modality, and the text modality. The findings of this ablation study are summarized in Table 4.

Our analysis shows that using a simple concatenation of the two modalities (denoted as "IOHunter w/o CrossAttn" in the table) is suboptimal—consistent with the results in Table 2—and offers limited advantages compared to a uni-modal classifier based solely on the graph modality (denoted as "IOHunter w/o Text" in the table). This aligns with prior findings on the challenges of training a multi-modal classifier that consistently outperforms the best uni-modal alternative (Wang, Tran, and Feiszli 2020). Moreover, incorporating the text modality via the cross-attention mechanism (denoted as "IOHunter" in the table) leads to an approximately 9% improvement over the graph-only approach. This result highlights the need to use both information sources for an effective IO classifier.

Conclusions

In this work, we introduced IOHunter, a novel GFM designed to detect IOs across various social media platforms. By integrating the strengths of LMs and GNNs, our approach leverages both textual and network-based features to accurately identify IO drivers across multiple countries and campaigns.

IOHunter significantly outperforms state-of-the-art methods in *supervised*, *scarcely-supervised*, and *cross-country* IO settings. This is achieved through the model's ability to generalize across diverse and evolving contexts, enabled by pretraining on extensive, multi-country datasets and fine-tuning with minimal labeled data. The robustness of IOHunter in scenarios with limited data availability underscores its practical applicability in real-world settings, where access to labeled data is often limited. Additionally, the cross-country generalization experiments highlight the model's potential to transfer knowledge between different geopolitical contexts.

Moving forward, our work paves the way for the development of more sophisticated GFMs tailored to more graph-related tasks. Future research could explore the application of IOHunter to domains where detecting coordinated, malicious activities is critical, as well as the integration of additional data modalities to further improve detection capabilities. Ultimately, IOHunter represents a significant step toward safeguarding the integrity of online discourse by providing a scalable, adaptable, and highly effective solution for uncovering information operations on social media platforms.

Acknowledgements

Work supported in part by DARPA (contract #HR001121C0169). MM acknowledges partial support by the SERICS project (PE00000014) under the NRRP MUR program funded by the EU - NGEU.

References

Abavisani, M.; Wu, L.; Hu, S.; Tetreault, J.; and Jaimes, A. 2020. Multimodal categorization of crisis events in social media. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14679–14689.

Alizadeh, M.; Shapiro, J. N.; Buntain, C.; and Tucker, J. A. 2020. Content-based features predict social media influence operations. *Science advances*, 6(30): eabb5824.

Badawy, A.; Ferrara, E.; and Lerman, K. 2018. Analyzing the digital traces of political manipulation: The 2016 Russian interference Twitter campaign. In *2018 IEEE/ACM in-*

ternational conference on advances in social networks analysis and mining (ASONAM), 258–265. IEEE.

Chen, Z.; and Subramanian, D. 2018. An Unsupervised Approach to Detect Spam Campaigns that Use Botnets on Twitter. arXiv:1804.05232.

Cherti, M.; Beaumont, R.; Wightman, R.; Wortsman, M.; Ilharco, G.; Gordon, C.; Schuhmann, C.; Schmidt, L.; and Jitsev, J. 2023. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2818–2829.

Cresci, S.; Di Pietro, R.; Petrocchi, M.; Spognardi, A.; and Tesconi, M. 2016. DNA-inspired online behavioral modeling and its application to spambot detection. *IEEE Intelligent Systems*, 31(5): 58–64.

Cui, H.; Lu, Z.; Li, P.; and Yang, C. 2022. On positional and structural node features for graph neural networks on non-attributed graphs. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 3898–3902.

Damianou, A.; Fabbri, F.; Gigioli, P.; De Nadai, M.; Wang, A.; Palumbo, E.; and Lalmas, M. 2024. Towards Graph Foundation Models for Personalization. In *Companion Proceedings of the ACM on Web Conference 2024*, 1798–1802.

De Nadai, M.; Fabbri, F.; Gigioli, P.; Wang, A.; Li, A.; Silvestri, F.; Kim, L.; Lin, S.; Radosavljevic, V.; Ghael, S.; et al. 2024. Personalized audiobook recommendations at spotify through graph neural networks. In *Companion Proceedings of the ACM on Web Conference 2024*, 403–412.

Deb, A.; Luceri, L.; Badaway, A.; and Ferrara, E. 2019. Perils and challenges of social media and election manipulation analysis: The 2018 us midterms. In *Companion proceedings of the 2019 world wide web conference*, 237–247.

Ezzeddine, F.; Ayoub, O.; Giordano, S.; Nogara, G.; Sbeity, I.; Ferrara, E.; and Luceri, L. 2023. Exposing influence campaigns in the age of LLMs: a behavioral-based AI approach to detecting state-sponsored trolls. *EPJ Data Science*, 12(1): 46.

Ferrara, E. 2015. Manipulation and abuse on social media. *ACM SIGWEB Newsletter*, (Spring): 1–9.

Ferrara, E. 2023. Social bot detection in the age of ChatGPT: Challenges and opportunities. *First Monday*, 28(6).

Fey, M.; and Lenssen, J. E. 2019. Fast graph representation learning with PyTorch Geometric. *arXiv preprint arXiv:1903.02428*.

Grinberg, N.; Joseph, K.; Friedland, L.; Swire-Thompson, B.; and Lazer, D. 2019. Fake news on Twitter during the 2016 US presidential election. *Science*, 363(6425): 374–378.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Hernandez, D.; Kaplan, J.; Henighan, T.; and McCandlish, S. 2021. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*.

Hristakieva, K.; Cresci, S.; Martino, G. D. S.; Conti, M.; and Nakov, P. 2022. The Spread of Propaganda by Coordinated Communities on Social Media. In *14th ACM Web Science Conference 2022*. ACM.

Huang, Q.; Ren, H.; Chen, P.; Kržmanc, G.; Zeng, D.; Liang, P. S.; and Leskovec, J. 2024. Prodigy: Enabling in-context learning over graphs. *Advances in Neural Information Processing Systems*, 36.

Im, J.; Chandrasekharan, E.; Sargent, J.; Lighthammer, P.; Denby, T.; Bhargava, A.; Hemphill, L.; Jurgens, D.; and Gilbert, E. 2020. Still out there: Modeling and identifying russian troll accounts on twitter. In *12th ACM Conference on Web Science*, 1–10.

Jiang, X.; Jia, T.; Fang, Y.; Shi, C.; Lin, Z.; and Wang, H. 2021. Pre-training on large-scale heterogeneous graph. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 756–766.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Kipf, T. N.; and Welling, M. 2022. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*.

Kong, Q.; Calderon, P.; Ram, R.; Boichak, O.; and Rizoiu, M.-A. 2023. Interval-censored transformer hawkes: Detecting information operations using the reaction of social systems. In *Proceedings of the ACM Web Conference 2023*, 1813–1821.

Lim, D.; Hohne, F.; Li, X.; Huang, S. L.; Gupta, V.; Bhalerao, O.; and Lim, S. N. 2021. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Advances in Neural Information Processing Systems*, 34: 20887–20902.

Liu, H.; Feng, J.; Kong, L.; Liang, N.; Tao, D.; Chen, Y.; and Zhang, M. 2024. One For All: Towards Training One Graph Model For All Classification Tasks. In *The Twelfth International Conference on Learning Representations*.

Lu, Y.; Jiang, X.; Fang, Y.; and Shi, C. 2021. Learning to pre-train graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 4276–4284.

Luceri, L.; Boniardi, E.; and Ferrara, E. 2024. Leveraging Large Language Models to Detect Influence Campaigns on Social Media. In *Companion Proceedings of the ACM on Web Conference 2024*, 1459–1467.

Luceri, L.; Deb, A.; Giordano, S.; and Ferrara, E. 2019. Evolution of bot and human behavior during elections. *First Monday*, 24(9).

Luceri, L.; Giordano, S.; and Ferrara, E. 2020. Detecting troll behavior via inverse reinforcement learning: A case study of Russian trolls in the 2016 US election. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, 417–427.

Luceri, L.; Pantè, V.; Burghardt, K.; and Ferrara, E. 2024. Unmasking the web of deceit: Uncovering coordinated activity to expose information operations on twitter. In *Proceedings of the ACM on Web Conference 2024*, 2530–2541.

Magelinski, T.; Ng, L.; and Carley, K. 2022. A synchronized action framework for detection of coordination on social media. *Journal of Online Trust and Safety*, 1(2).

Mannocci, L.; Mazza, M.; Monreale, A.; Tesconi, M.; and Cresci, S. 2024. Detection and Characterization of Coordinated Online Behavior: A Survey. *arXiv preprint arXiv:2408.01257*.

Mao, H.; Chen, Z.; Tang, W.; Zhao, J.; Ma, Y.; Zhao, T.; Shah, N.; Galkin, M.; and Tang, J. 2024. Graph foundation models. *arXiv preprint arXiv:2402.02216*.

Mazza, M.; Avvenuti, M.; Cresci, S.; and Tesconi, M. 2022. Investigating the difference between trolls, social bots, and humans on Twitter. *Computer Communications*, 196: 23–36.

Minici, M.; Cinus, F.; Luceri, L.; and Ferrara, E. 2024. Uncovering coordinated cross-platform information operations: Threatening the integrity of the 2024 U.S. presidential election. *First Monday*, 29(11).

Nizzoli, L.; Tardelli, S.; Avvenuti, M.; Cresci, S.; and Tesconi, M. 2021. Coordinated behavior on social media in 2019 UK general election. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 15, 443–454.

Nogara, G.; Vishnuprasad, P. S.; Cardoso, F.; Ayoub, O.; Giordano, S.; and Luceri, L. 2022. The Disinformation Dozen: An Exploratory Analysis of Covid-19 Disinformation Proliferation on Twitter. In *14th ACM Web Science Conference 2022*, 348–358.

Nwala, A. C.; Flammini, A.; and Menczer, F. 2023. A language framework for modeling social media account behavior. *EPJ Data Science*, 12(1): 33.

Pacheco, D.; Flammini, A.; and Menczer, F. 2020. Unveiling Coordinated Groups Behind White Helmets Disinformation. In *Companion Proceedings of the Web Conference 2020*. ACM.

Pacheco, D.; Hui, P.-M.; Torres-Lugo, C.; Truong, B. T.; Flammini, A.; and Menczer, F. 2021. Uncovering Coordinated Networks on Social Media: Methods and Case Studies. *Proceedings of the International AAAI Conference on Web and Social Media*, 15(1): 455–466.

Pozzana, I.; and Ferrara, E. 2020. Measuring bot and human behavioral dynamics. *Frontiers in Physics*, 8(125).

Qiu, J.; Chen, Q.; Dong, Y.; Zhang, J.; Yang, H.; Ding, M.; Wang, K.; and Tang, J. 2020. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 1150–1160.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Seckin, O. C.; Pote, M.; Nwala, A.; Yin, L.; Luceri, L.; Flammini, A.; and Menczer, F. 2024. Labeled Datasets for Research on Information Operations. *arXiv preprint arXiv:2411.10609*.

Shao, C.; Ciampaglia, G. L.; Varol, O.; Yang, K.-C.; Flammini, A.; and Menczer, F. 2018. The spread of low-credibility content by social bots. *Nature communications*, 9(1): 1–9.

Sharma, K.; Zhang, Y.; Ferrara, E.; and Liu, Y. 2021. Identifying Coordinated Accounts on Social Media through Hidden Influence and Group Behaviours. In *KDD '21: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

Starbird, K. 2019. Disinformation's spread: bots, trolls and all of us. *Nature*, 571(7766): 449–450.

Stella, M.; Ferrara, E.; and De Domenico, M. 2018. Bots increase exposure to negative and inflammatory content in online social systems. *Proceedings of the National Academy of Sciences*, 115(49): 12435–12440.

Suresh, V. P.; Nogara, G.; Cardoso, F.; Cresci, S.; Giordano, S.; and Luceri, L. 2024. Tracking Fringe and Coordinated Activity on Twitter Leading Up To the US Capitol Attack. In *Proceedings of the International AAAI Conference on Web and Social Media*.

Wang, W.; Tran, D.; and Feiszli, M. 2020. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12695–12705.

World Economic Forum. 2024. The Global Risks Report 2024. https://www.weforum.org/publications/global-risks-report-2024/.

Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; and Philip, S. Y. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1): 4–24.

Xie, H.; Zheng, D.; Ma, J.; Zhang, H.; Ioannidis, V. N.; Song, X.; Ping, Q.; Wang, S.; Yang, C.; Xu, Y.; et al. 2023. Graph-aware language model pre-training on a large graph corpus can help multiple graph applications. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 5270–5281.

Yang, K.-C.; Ferrara, E.; and Menczer, F. 2022. Botometer 101: Social bot practicum for computational social scientists. *Journal of computational social science*, 5: 1511–1528.

Yang, K.-C.; Varol, O.; Davis, C. A.; Ferrara, E.; Flammini, A.; and Menczer, F. 2019. Arming the public with artificial intelligence to counter social bots. *Human Behavior and Emerging Technologies*, 1(1): 48–61.

Zannettou, S.; Caulfield, T.; Cristofaro, E. D.; Sirivianos, M.; Stringhini, G.; and Blackburn, J. 2019. Disinformation Warfare: Understanding State-Sponsored Trolls on Twitter and Their Influence on the Web. arXiv:1801.09288.

Zhai, X.; Kolesnikov, A.; Houlsby, N.; and Beyer, L. 2022. Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12104–12113.

Zheng, X.; Wang, Y.; Liu, Y.; Li, M.; Zhang, M.; Jin, D.; Yu, P. S.; and Pan, S. 2022. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082*.