T-NET: Weakly Supervised Graph Learning for Combatting Human Trafficking

Pratheeksha Nair[1,2], Javin Liu[2], Catalina Vajiac[3], Andreas Olligschlaeger[4], Duen Horng Chau[5], Mirela Cazzolato[6], Cara Jones[7], Christos Faloutsos[3], Reihaneh Rabbany[1,2]

[1]McGill University
[2]Mila - Quebec AI Institute
[3]Carnegie Mellon University
[4]i3 LLC
[5]Georgia Institute of Technology
[6]University of Sao Paulo
[7]Marinus Analytics
pratheeksha.nair@mail.mcgill.ca

Abstract

Human trafficking (HT) for forced sexual exploitation, often described as modern-day slavery, is a pervasive problem that affects millions of people worldwide. Perpetrators of this crime post advertisements (ads) on behalf of their victims on adult service websites (ASW). These websites typically contain hundreds of thousands of ads including those posted by independent escorts, massage parlor agencies and spammers (fake ads). Detecting suspicious activity in these ads is difficult and developing data-driven methods is challenging due to the hard-to-label, complex and sensitive nature of the data.

In this paper, we propose T-NET, which unlike previous solutions, formulates this problem as weakly supervised classification. Since it takes several months to years to investigate a case and obtain a single definitive label, we design domain-specific signals or indicators that provide weak labels. T-NET also looks into connections between ads and models the problem as a graph learning task instead of classifying ads independently. We show that T-NET outperforms all baselines on a real-world dataset of ads by 7% average weighted F1 score. Given that this data contains personally identifiable information, we also present a realistic data generator and provide the first publicly available dataset in this domain which may be leveraged by the wider research community.

Introduction

Human Trafficking (HT), defined as the recruitment, transportation and control of persons, typically through sexual exploitation and forced labour (Canada 2023), is a problem that affects over 6.3 million people worldwide in a year (Organization 2022). In majority of the cases, advertisements (ads) are posted by traffickers on behalf of their victims on adult service websites (ASW) (Thorn 2015; Crotty and Bouché 2018) and an average pimp has control over 4 to 6 victims; many of whom report of having “no input into the wording used in the advertisements” (Thorn 2015). For example, in a convicted case in British Columbia in 2014 (R. v. Moazami), the trafficker solicited to illegally push 9 underage girls and 2 women into the trafficking industry advertising them through various escort websites. These websites provide easy-to-use and low-risk platforms for the traffickers, allowing them to remain anonymous and operate on a wide geographical range. Thus, monitoring these online escort markets is the key to combating human trafficking.

Generally, law enforcement officers scour through thousands of these online ads manually looking for suspicious activity (e.g, keywords indicative of underage activity). They then collect evidences and look for other connected ads and phone numbers which is a tedious and time-consuming process. Thus, finding clues becomes a needle-in-the-haystack problem making manual investigation very difficult.

To address this issue, text clustering and ranking based approaches have been proposed (Kulshrestha 2021; Lee et al. 2021) which not only make it easier to analyze the ads but also facilitate their interactive visualizations (Nair et al. 2022; Vajiac et al. 2022). These approaches are based on one critical insight on HT detection that traffickers tend to almost entirely control the ad content for their victims. Consequently, ads posted by the same trafficker for multiple victims tend to be very similar (U.N 2021; Project 2014). However, these websites also contain ads posted by independent, at-will sex workers and separating the two is non-trivial as they look more-or-less the same. Moreover, there exist other types of behaviour, or *modus operandi* (M.Os) as we call it, such as massage parlors and spam – ads with false contact information.

The exact motivation for spam is not clear but one hypothesis is that sometimes well-intentioned parties inject fake ads into websites to discourage buyers from purchasing sexual services. Another hypothesis is that traffickers may pump fake ads into these websites to throw off investigators and stay under the radar. Spam adds noise to the data affecting both manual investigation as well as machine learning approaches. Automating the identification of these activities is also challenging as labels are hard to acquire. Current methods (Lee et al. 2021; Kulshrestha 2021) are helpful in finding clustered activities but they do not characterize or provide insights on the different type of M.Os.

Previous works on HT detection (Alvari, Shakarian, and Snyder 2017; Dubrawski et al. 2015; Tong et al. 2017) were based on binary classification of ads into HT vs non-HT. These are limited in that 1) they study ads individually and potentially overlook important clues based on *similarities* and *connections* among ads and 2) they disregard other M.Os such as spam, which can add significant noise and independent sex work (ISW), which when conflated with HT is harmful; 3) their methods are not open-source and were developed on private datasets, labelled using domain experts, making it difficult to adopt and reproduce.

In this paper, we address the above limitations through T-NET that 1) classifies clusters instead of individual ads while leveraging connections between them, 2) models the problem as multi-class classification and detects different types of activities and 3) is an open-source solution that uses *labeling functions* (LFs) to programmatically map the knowledge and heuristics used by expert labelers to weak labels. We also release ASW-SYNTH, a synthetically generated corpus of realistic escort ads. To summarize, the main contributions of this work are three-fold.

- **Novel problem formulation.** Unlike previous works, we tackle the problem of HT detection in a novel way by finding similarity-based clusters of ads and leveraging the connections between them to model a graph learning problem.
- **Effective method.** We introduce T-NET, a new framework that combines graph contrastive learning and node classification in the presence of weak labels and show its effectiveness on two escort ads datasets with 7% improvement over baselines.
- **Reproducibility.** We introduce ASW-SYNTH, the first publicly shareable corpus of escort ads, facilitating further research in this field. We also provide *labeling functions* for the weak labeling of such datasets and our solution are publicly available.[1]

Related Work

In this section we discuss some background and related works to help motivate the problem of HT detection and our approach based on contrastive graph learning and weak supervision.

Human Trafficking Several strategies have been explored in the past decade to develop data-driven techniques for tracking online HT. These included usage of knowledge graphs (Szekely et al. 2015), text processing models (Dubrawski et al. 2015; Alvari, Shakarian, and Snyder 2017; Tong et al. 2017), information extraction (Nagpal et al. 2017), detecting ads authored by the same person and linking to bitcoin wallets (Portnoff et al. 2017) and analyzing images posted in ads (Stylianou, Souvenir, and Pless 2019). All these methods function at an ad level and predict the likelihood of individual ads being involved in trafficking and do not differentiate different types of activities within the market.

Graph Contrastive Learning Contrastive learning (CL) in graphs is a self-supervised approach that focuses on pulling a node and a *positive* sample closer to each other in the embedding space, while pushing it away from its *negative* samples (Khosla et al. 2020; Chen et al. 2020). For most graph CL (GCL) methods, node and graph level augmentations are contrasted in different ways typically between one or more augmented views of the graph (You et al. 2020; Hassani and Khasahmadi 2020; Zhu et al. 2020). GCL has also been explored in the context of deep graph clustering (Park et al. 2022) which showed that explicitly considering the network structure in the contrastive loss made the node embeddings more aligned to their respective class labels.

Graph Learning with Weak Labels When there are multiple weak labels associated with each data sample (like in our problem setting), also known as programmatic weak supervision (Ratner et al. 2016, 2019), majority vote (MV) is the simplest and most straight-forward strategy for label aggregation which chooses a label based on consensus from all the weak labelers. However, these MV aggregated labels are still noisy. Most common approaches for neural networks to deal with noisy labels are data-driven (Van Rooyen and Williamson 2017), learning objective (Reed et al. 2014) or optimization based (Arpit et al. 2017). PI-GNN (Du et al. 2023) is a recent work that introduces an adaptive noise estimation technique leveraging pairwise interactions between nodes for model regularization. NRGNN (Dai, Aggarwal, and Wang 2021) is another recent work that utilizes edge prediction to predict links between unlabelled and labelled nodes and expands the training set with pseudo labels, making it more robust to label noise. We compare T-NET with the most recent NRGNN and PI-GNN baselines.

Problem Formulation

Consider a graph $G = (V, A, X)$, with set of nodes V, adjacency matrix A and node feature matrix $X \in \mathbb{R}^{|V| \times d}$ of some dimension d. Each node V_i, which represents a cluster of ads, has an unobserved true label $y_i \in \{1, 2 \ldots C\}$ belonging to one of C classes. The list of all labels is denoted by $Y = [y_1, y_2 \ldots y_{|V|}]$. There are m labeling functions (LFs) applied on a cluster/node V_i to output $\dot{y}_i \in \{-1, 1, 2 \ldots C\}$ which can be aggregated using majority vote to get $\tilde{Y}_i$. -1 indicates abstain/no information on label. The goal of *weakly supervised node classification* is then to learn a node classifier model $f : G, \Lambda \rightarrow Y$ such that it uses the graph G and weak node labels Λ, to predict Y.

Proposed Method: T-NET

We introduce T-NET, a 3-layer GNN that consists of a classification component and a contrastive learning component.

Classification Component Consists of two graph convolution layers (GCONV) followed by softmax and aims to optimize the cross-entropy loss between the predicted label $\hat{Y}$ and aggregated label $\tilde{Y}$ while handling label noise through *weighted classification loss*.

T-NET uses Algorithm 1 to weigh the classification loss per node based on the uncertainty of its weak labels and how influential it is. In particular, for a given node, if most LFs agree on its label i.e, the entropy of the LF outputs for that node is small, we can be more certain of that label and give

[1] Our code base is available at https://github.com/nair-p/T-Net

that node a higher weight. The inverse is also true. Secondly, if its node embedding (h_i) is closer to a cluster center, then it has a higher influence on nearby embeddings and hence should be weighed accordingly. To capture this, the node embeddings (which represents clusters of ads) are further grouped using KMeans (with $k = C$) and if Q_i is the cluster that node i belongs to and q_i represents its cluster centroid, the centrality of node i is calculated as:

$$\rho_i = cent(i) = |Q_i| * \frac{(h_i.h_{q_i})}{\sum_{j=1}^{|V|}(h_j.h_{q_j})} \tag{1}$$

where '.' indicates the cosine similarity operator.

Contrastive learning component To further improve robustness of learned embeddings and reduce corruption by label noise, T-NET employs a contrastive learning (CL) component. The CL objective function aims to maximize the mutual information between a node and the graph community it belongs to. Groups of nodes in the same graph community are densely connected to each other and loosely connected to nodes from other communities. Due to network homophily, we know that nodes with a similar graph structure are more likely to belong to the same class. Thus, for finding a given node's positive sample, we randomly pick a node belonging to the same network community, computed using Louvain algorithm[2] (Blondel et al. 2008). For negative samples, we use a network corruption setup where the rows of X are shuffled, maintaining the same structure, and r negative rows are sampled uniformly at random. The contrastive loss function L_S is given by Equation 2 and the CL component process is described in Algorithm 2.

$$\mathrm{CL}(h_i, h_i^+, S^-) = -\log \frac{\exp(h_i.h_i^+)}{\sum\limits_{j=1}^{r} \exp(h_i.S_j^-) + \exp(h_i.h_i^+)} \tag{2}$$

where h_i^+ is the positive sample for node i and S^- is the set of its r negative samples. This CL component employs a 2-layer GNN to learn node embeddings. The first layer, AGG, is a mean aggregation of neighbors as:

$$\textsc{Agg}(x_i, \mathcal{N}_i) = ReLU(\mathbf{W}.\textsc{avg}(\{x_v | \forall v \in \mathcal{N}_i\})) \tag{3}$$

where $\mathcal{N}_i$ denotes the neighbors of node i, $\mathbf{W}$ represents the weights learned by the model and AVG denotes the average aggregation. The second layer, GCONV, is a classic GCN layer.

Algorithm 3 describes T-NET and shows how the classification component and a contrastive learning component come together. The final loss is a combination of the losses from these two components which is then backpropagated through the model. Figure 1 provides a visual illustration of the components in T-NET.

Algorithm 1: CLASSIFICATION-LOSS

```
Input: $\hat{Y}$, $\Lambda$, $C$
Output: $L_C$
$L \leftarrow emptyList(size = |\hat{Y}|)$
$m \leftarrow |\Lambda_0|$
for $i \in \{0 \ldots |\hat{Y}|\}$ do
    $\lambda \leftarrow zerolist(size = C)$                ▷ Majority Vote
    $\hat{y_i} \leftarrow \hat{Y_i}$
    for $j \in [1, .., m]$ do
        $\lambda[\Lambda_{ij}] \leftarrow \lambda[\Lambda_{ij}] + 1$
    end for
    $\tilde{y} \leftarrow arg\,max(\lambda)$
    $l \leftarrow CrossEntropy(\hat{y_i}, \tilde{y})$
    $e \leftarrow entropy(\lambda)$
    $\rho \leftarrow cent(i)$                          ▷ using Equation 1
    $nw \leftarrow e * \rho$
    $L[i] \leftarrow l * nw$
end for
$L_C \leftarrow \frac{1}{|L|} \sum_{j=1}^{|L|} L_j$
return $L_C$
```

Algorithm 2: CONTRASTIVE-LOSS

```
Input: $H$, $V$, $X$, $A$, $Comms$
Hyperparameter: $r$
Output: $L_S$
$L \leftarrow emptyList(size = |H|)$
for $i \in \{0 \ldots |H|\}$ do
    $s^+ \sim Comms(V_i)$  ▷ sample a node from community of node $i$
    $h_i^+ \leftarrow H_{s^+}$
    $\tilde{X} \leftarrow Shuffle\_rows(X)$
    $S^- \leftarrow emptyList(size = r)$
    Let $j = 1$
    while $j \leq r$ do
        $\tilde{x} \sim \tilde{X}$                  ▷ sample from shuffled $X$
        $\tilde{z} \leftarrow \textsc{agg}(\tilde{x}, A)$
        $\tilde{h} \leftarrow \textsc{gconv}(\tilde{z})$
        $S^-[j] \leftarrow \tilde{h}$
        $j \leftarrow j + 1$
    end while
    $L[i] \leftarrow \mathrm{CL}(h_i, h_i^+, S^-)$          ▷ using Equation 2
end for
$L_S \leftarrow \frac{1}{|L|} \sum_{j=1}^{|L|} L_j$
return $L_S$
```

Datasets

ASW-REAL: Real-World Dataset

ASW-REAL (Adult Service Website-Real) is an escort ads datasets of 10199 ads with HT, Spam and ISW labels manually labelled and provided to us by our collaborators working in this field. It consists of ad text and structured selectors (such as advertised aliases, location and date of posting, contact information) maskable using unique hash values.

[2] Here we merely demonstrate using simple Louvain method and note that any community detection algorithm may be used.

Algorithm 3: T-Net

```
Input: $V$, $A$, $X$, $\Lambda$, $C$
Hyperparameter: $T$
Output: $\hat{Y}$
Let $t = 1$
$Comms \leftarrow Louvain(A)$   ▷ obtain graph communities
while $t \leq T$ do
    $Z_1 \leftarrow \text{AGG}(X, A)$   ▷ using Equation 3
    $H \leftarrow \text{GCONV}(Z_1)$   ▷ GCN layer
    $Z_2 \leftarrow \text{GCONV}(H)$
    $\hat{Y} \leftarrow softmax(Z_2)$
    $L_C \leftarrow \text{CLASSIFICATION-LOSS}(\hat{Y}, H, \Lambda, C)$
    $L_S \leftarrow \text{CONTRASTIVE-LOSS}(H, V, X, A, Comms)$
    $L \leftarrow L_S + L_C$
    $t \leftarrow t + 1$
    Backpropagate $L$
end while
return $\hat{Y}$
```

ASW-Synth: Synthetically Generated Dataset

Escort ads often contain sensitive information, explicit language and images, and personally identifiable information such as phone numbers and social media tags. Although these are publicly accessible online, there are constraints to sharing such datasets. To address this, we produce ASW-Synth, a synthetically generated corpus of realistic escort ads that is shareable and can be leveraged by the broader research community.

The escort ad content in ASW-Synth were created using the state-of-the-art language model, gpt-3.5-turbo. Relevant meta-data such as phone numbers, email addresses, location of posting, social media tags and posting date of ads, were then added. Following this, clusters were manually included by the duplication of ad text and connections were forged between them by means of shared meta-data. Lastly, 3 types of M.Os were inserted based on their corresponding indicators. This workflow is further explained below.

Ad text generation Using the OpenAI ChatCompletion API with the gpt-3.5-turbo model, we generated 7236 unique ads. The prompt used consisted of 1) *background information* about HT in escort ads, 2) *generation specification* in terms of the required format, tone and style of the ads such as mentions of advertised aliases, 3) *4 anonymized examples* from real-world escort ads and 4) *feedback* on current generations for improvement.

Adding meta-data information For each of the generated ads, a random posting date between January and July 2022 was added. A posting location was randomly selected from a given list of Canadian city names. Phone numbers were randomly generated (fake numbers) and masked using MD5-hashing. Advertised names were extracted from the ads using a domain-specific name extractor (Li et al. 2022) and email addresses and social media tags were created by adding tags to the names, such as *@mail.com* and *_xoxo* respectively.

Dataset	Nodes	Edges	Comms.	Comps.
ASW-Real	438	28092	17	12
ASW-Synth	7236	2951206	722	721

Table 1: Statistics of the constructed graphs for the real-world and synthetic datasets. Comms. is the number of detected communities and Comps. represents the number of connected components.

Adding clusters and connections Each of the generated ads acted as cluster center (hence 7236 clusters) and was duplicated as is or after removing 1-5% of words to simulate near-duplicates. Cluster sizes were sampled from a pareto distribution with a minimum size of 100 and maximum size of 50000. As a result, there are few large clusters and several small ones around 100-200 ads resulting in a corpus of around 868279 ads. For adding connections, pairs of clusters were randomly sampled and assigned the same phone number, email addresses and/or social media tags resulting in around 2.9 million connections.

Inserting different types of M.Os For Spam, we randomly selected clusters and added over 10 posting locations and 5 phone numbers. Their posting pattern was made "bursty" where a large volume of ads get posted on one day followed by no posts for several days to emulate the outcome of automated bots/scripts. For HT, we randomly inserted certain keywords (according to HT LFs in Table 3) into the ad text and ensured that at least 3 or more persons were advertised within the same cluster. For ISW, we included relevant keywords into the ads and ensured each cluster had only 1-2 persons advertised with 1-2 phone numbers to indicate one or two independent escorts. The data statistics for ASW-Synth are provided in Table 1.

Methodology Pipeline

For both ASW-Synth and ASW-Real, the below pipeline (also displayed in Figure 1) was used for preparing the dataset, building the graph and obtaining suspicious clusters.

Preprocessing Any ads with duplicate information were removed (more relevant for ASW-Real). We used the state-of-the-art entity extractor for this domain (NEAT) (Li et al. 2022) to obtain the aliases mentioned in the ads. This is useful in determining how many individuals are potentially being advertised in the same ad or group of ads.

Grouping related ads We then clustered the ads based on text similarity using InfoShield (Lee et al. 2021), the state-of-the-art clustering algorithm for the HT domain to create text-based clusters (called micro-clusters).

Constructing the graph Having obtained micro-clusters of related ads, we look for shared information (called meta-data) among these clusters. More specifically, the clusters are treated as the nodes in a graph and there exists an edge between two nodes if they share any hard links such as a phone number, email address or social media tag. This allows us to build a network of clusters based on common

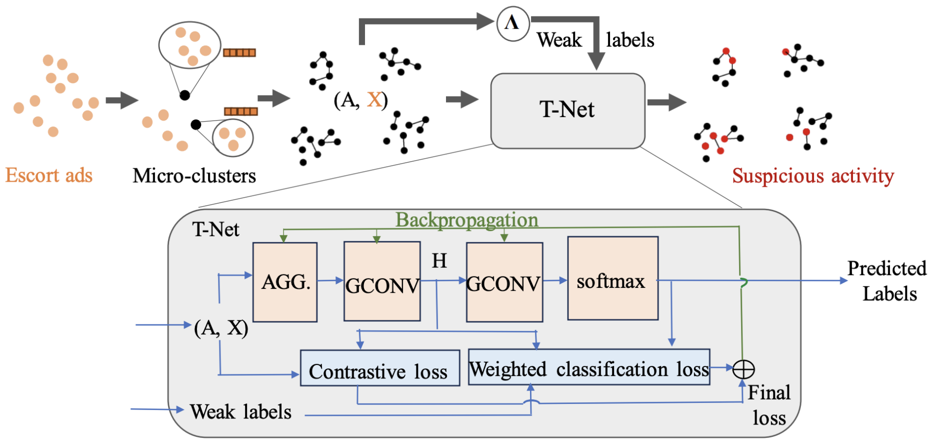

Figure 1: The methodology pipeline. Escort ads are clustered based on text similarity to obtain micro-clusters, which form the nodes. Micro-cluster nodes are linked to each other based on shared meta-data (e.g. shared phone number) to build a graph. Node features are extracted and the LFs are applied on the clusters to get weak labels (Λ). The graph (A) node features (X) and Λ are input to T-NET and the nodes predicted as HT are considered to be suspicious micro-clusters.

meta-data. The clusters are also characterized by features as defined in Table 2. Each of these represent one value in a vector which is then treated as a node feature.

Feature	**Description**
Cluster Size	Number (#) of ads
Phone Count	# of unique phone numbers
Phone Entropy	Entropy of phone number counts
Email Count	# of unique email IDs
Location Count	# of distinct locations
Location Radius	Radius of ad posting location
Social Media Count	# of social media accounts
URL Count	# of total URLs in the ad
Valid URL Count	# of valid URLs in the ad
Invalid URL Count	# of invalid URLs in the ad
Advertised Aliases	# of persons advertised in a cluster
Ads per week	# of ads posted per week

Table 2: List of features used for cluster characterization.

Obtaining weak labels Over the course of discussions with domain experts, criminologists and survivors, and based on related works in the literature(L'Hoiry, Moretti, and Antonopoulos 2021), we defined several labeling functions/LFs (referred to as Λ)[3] as described in Table 3.

These LFs look for the presence of certain indicators and count the number of their occurrences within a micro-cluster. We then plot their distribution and identify inflection points (the value at which there is a sudden change in the distribution curves) which are calculated as the last point where the sign of the second derivative changes. These points are then set as thresholds for the corresponding LFs and all micro-clusters with LF values above the threshold are assigned to the corresponding class. For example, for 'Number of advertised aliases', the threshold was set at 4 and for micro-clusters with 4 or more persons advertised, it returns HT as the label.

For ASW-REAL, the average coverage (% of ads which do not get a -1 label) is 10.7% and the average accuracy is 53%. For ASW-SYNTH, these values are 20% and 67% respectively. These values indicate the quality of the *labeled* data available for this task and further demonstrates the need for our weakly supervised classification pipeline.

Getting suspicious clusters The constructed graph of micro-clusters (V, A), node features(X) and weak label matrix (Λ) are input to T-NET which is trained using majority-vote aggregated weak labels of the training nodes. The predictions on a test-set are obtained as per Algorithm 3 and evaluated using the F_β score defined by Equation 4. This is repeated for 5 random train-test splits.

$$F_\beta = (1+\beta^2)\frac{precision \times recall}{(\beta^2 \times precision) + recall} \tag{4}$$

We report the mean and standard deviation of F_β scores for each of the 3 classes (HT, ISW and Spam) in Tables 4 and 5. $\beta = 1$ is the standard F-1 measure where precision and recall get equal weights. When $\beta = 0.5$, precision is given higher weight than recall. False positives for HT are more harmful than true negatives, making this a reasonable choice of evaluation metric.

Experiments and Results

Baselines: We compare T-NET with the following baselines and variations of the method as ablations.

- MLP: Multi-Layer Perceptron classifier that disregards connections between nodes and treats them as independent data points.

[3] The keywords used in LFs 1-4 and 7-9 are similar to the ones used in (L'Hoiry, Moretti, and Antonopoulos 2021) and are provided in https://github.com/nair-p/T-Net/blob/main/labeling_functions.py.

			ASW-Real		ASW-Synth	
	Labelling Function	**Description**	**cov.**	**acc.**	**cov.**	**acc.**
HT	HT Keywords	presence of suspicious keywords	25.6	20.5	9.53	74.7
	No restrictions in service	willing to do any the kind of service	13.9	3.30	14.87	48.1
	Incall only/No outcall	indicates restrictions in movement	0.20	100	8.98	27.2
	3^{rd} / 1^{st} person plural pronouns	advertising someone other than themselves	6.40	32.1	5.14	62.1
	Advertised aliases	multiple people being advertised	24.7	6.50	28.0	24.1
	Image ids per phone number	multiple image ids linked to the same phone	28.3	85.5	x	x
ISW	Non-HT Keywords	keywords associated with ISW	13.5	94.9	11.6	100.
	Restrictions in services	based on sex worker preferences	7.30	93.7	12.1	100.
	Availability without restrictions	no restrictions in movement	0.20	100	7.02	100.
Spam	Phone number count	either 0 or a large number of phone numbers	3.42	33.3	6.72	61.7
	Post location radius	ads spread out in a very short timeframe	2.30	90.0	14.7	85.8
	Number of locations	ads spread out across the country	2.70	75.0	16.9	100.

Table 3: List of LFs defined for the 3 classes with their individual coverage and accuracy on each of the datasets. We can see that the coverage of the individual LFs are relatively low which suggest the need to aggregate them to achieve better average coverage. Please note that ASW-Synth does not contain images, hence "Image ids per phone number" is not applied on it.

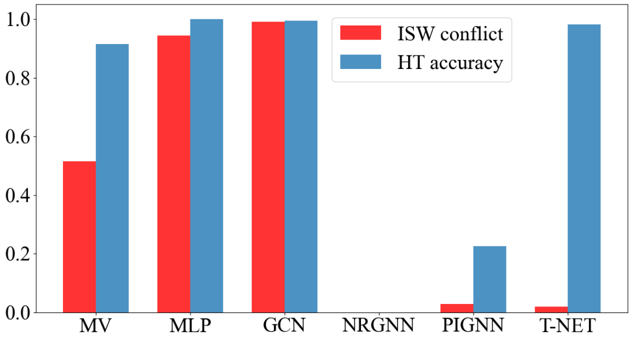

Figure 2: T-Net accurately detects HT while not conflicting with ISW. Higher values of HT accuracy and lower values of ISW conflict are better. MV is the majority vote aggregated labels used for training the models. NRGNN classifies everything as ISW and has both values zero.

- GCN: A 2 layer Graph Convolution Network (Kipf and Welling 2016) that does not explicitly handle label noise.
- NRGNN (Dai, Aggarwal, and Wang 2021): Noise-Resistant Graph Neural Network that augments the graph using edge prediction and expands the labelled training set with pesudo labels.
- PI-GNN (Du et al. 2023): Method that regularizes the model to noise by leveraging similarities between pairs of nodes.
- T-Net–CL: Our method without the contrastive-loss component.
- T-Net–nw: Our method whaere all node-weights are 1.
- T-Net–E: Our method where the node weights do not consider the entropy of the weak labels.
- T-Net–ρ: Our method where the node weights do not consider the centrality of node embeddings.

Classification results: In Tables 4 and 5, we see that T-Net outperforms all baselines on average and has a clear advantage on the HT and ISW classes. More specifically, in the overall weighted average score, T-Net's F1 score is 7% and 23% better than the next best baseline on ASW-Real and ASW-Synth respectively. We observe that PI-GNN does better on the Spam class as its loss function is equipped to handle sample imbalance (ASW-Real has only 9 spam examples) and disconnected nodes in the data. Spam nodes are generally not connected to multiple other nodes due to lack of connecting meta-data and as a result, are either isolated or belong to small, disconnected communities.

Confusion of sensitive classes: It is important to ensure that false positives of the HT class are minimized, especially when they belong to ISW. These false positives can result in harm to vulnerable communities (discussed more in the societal impact), in addition to wasting the time of investigators and reducing the trust in the system. We study this aspect by measuring the *ISW conflict* – proportion of ISW nodes misclassified as HT. In Figure 2, we plot the ISW conflict along with HT accuracy for different methods on ASW-Synth and show that T-Net has the lowest ISW conflict while maintaining the highest HT accuracy. A similar pattern is also observed on the ASW-Real dataset.

Discussion and Conclusions

Societal impact Narratives around HT have often been conflated with sex work as highlighted by studies (Durisin and van der Meulen 2021) and most previous machine-learning based approaches to HT detection, do not explicitly handle differentiating between these two types of activities. This can be harmful for at-will sex workers through over policing, unhelpful interventions, oversurveillance, displacing work venues, etc. Unlike previous works, we have considered these two activities separately and designed specific LFs for both. Moreover, since our analysis is at a cluster level, our focus is more on large, connected clusters without targeting individual ads. Unlike previous works (Tong et al. 2017), T-Net is not designed based on any physical

Method	**HT**		**ISW**		**Spam**		**Weighted Average**	
	$\beta = 0.5$	$\beta = 1$	$\beta = 0.5$	$\beta = 1$	$\beta = 0.5$	$\beta = 1$	$\beta = 0.5$	$\beta = 1$
MLP	0.57 ± 0.15	0.62 ± 0.06	0.08 ± 0.02	0.04 ± 0.04	0.06 ± 0.50	0.10 ± 0.15	0.45 ± 0.07	0.35 ± 0.03
GCN	0.68 ± 0.01	0.77 ± 0.06	0.38 ± 0.10	0.23 ± 0.16	0.21 ± 0.45	0.24 ± 0.25	0.64 ± 0.06	0.52 ± 0.10
NRGNN(2021)	0.50 ± 0.38	0.56 ± 0.27	0.00 ± 0.00	0.01 ± 0.02	0.25 ± 0.46	0.23 ± 0.36	0.33 ± 0.21	0.31 ± 0.15
PI-GNN (2023)	0.74 ± 0.01	0.82 ± 0.07	0.70 ± 0.07	0.51 ± 0.10	**0.29 ± 0.30**	**0.28 ± 0.28**	0.77 ± 0.06	0.67 ± 0.09
T-Net–CL	0.73 ± 0.01	0.80 ± 0.07	0.49 ± 0.22	0.30 ± 0.11	0.11 ± 0.08	0.14 ± 0.12	0.71 ± 0.13	0.57 ± 0.08
T-Net–nw	0.75 ± 0.13	0.81 ± 0.07	0.53 ± 0.40	0.35 ± 0.22	**0.29 ± 0.37**	0.23 ± 0.22	0.71 ± 0.16	0.59 ± 0.12
T-Net–E	0.74 ± 0.01	0.81 ± 0.07	0.78 ± 0.02	0.56 ± 0.20	0.17 ± 0.45	0.20 ± 0.17	**0.82 ± 0.06**	0.69 ± 0.13
T-Net–ρ	0.67 ± 0.36	0.69 ± 0.30	0.68 ± 0.20	0.43 ± 0.27	0.27 ± 0.38	0.19 ± 0.23	0.71 ± 0.12	0.59 ± 0.16
T-Net	**0.77 ± 0.15**	**0.83 ± 0.10**	**0.85 ± 0.02**	**0.66 ± 0.20**	0.11 ± 0.40	0.13 ± 0.27	**0.82 ± 0.13**	**0.74 ± 0.16**

Table 4: T-Net performs better than baselines on ASW-Real. We report 5-fold cross-validation F_β results. ASW-Real contains 231 HT, 198 ISW and 9 Spam clusters. Highest values are in bold and second highest are in red.

Method	**HT**		**ISW**		**Spam**		**Weighted Average**	
	$\beta = 0.5$	$\beta = 1$	$\beta = 0.5$	$\beta = 1$	$\beta = 0.5$	$\beta = 1$	$\beta = 0.5$	$\beta = 1$
MLP	0.22 ± 0.07	0.30 ± 0.01	0.05 ± 0.02	0.02 ± 0.05	**0.99 ± 0.00**	**0.98 ± 0.00**	0.45 ± 0.03	0.26 ± 0.03
GCN	0.38 ± 0.08	0.43 ± 0.30	0.54 ± 0.45	0.51 ± 0.42	0.72 ± 0.39	0.73 ± 0.38	0.55 ± 0.29	0.55 ± 0.27
NRGNN(2021)	0.17 ± 0.18	0.23 ± 0.16	0.39 ± 0.40	0.18 ± 0.36	0.76 ± 0.30	0.76 ± 0.38	0.48 ± 0.20	0.31 ± 0.15
PI-GNN (2023)	0.21 ± 0.01	0.31 ± 0.01	0.20 ± 0.40	0.10 ± 0.09	0.97 ± 0.02	**0.98 ± 0.01**	0.57 ± 0.26	0.30 ± 0.06
T-Net–CL	**0.59 ± 0.31**	0.46 ± 0.30	0.89 ± 0.06	0.76 ± 0.22	0.94 ± 0.02	0.79 ± 0.39	**0.87 ± 0.03**	0.73 ± 0.24
T-Net–nw	0.26 ± 0.36	0.30 ± 0.14	0.86 ± 0.04	0.66 ± 0.34	0.69 ± 0.37	0.65 ± 0.36	0.77 ± 0.07	0.62 ± 0.19
T-Net–E	0.29 ± 0.17	0.31 ± 0.25	0.76 ± 0.20	0.65 ± 0.27	0.69 ± 0.38	0.66 ± 0.36	0.67 ± 0.21	0.61± 0.22
T-Net–ρ	0.44 ± 0.29	0.44 ± 0.27	0.88 ± 0.12	0.83 ± 0.09	0.72 ± 0.38	0.69 ± 0.39	0.79 ± 0.13	0.75 ± 0.11
T-Net	**0.59 ± 0.28**	**0.64 ± 0.22**	**0.95 ± 0.03**	**0.90 ± 0.11**	0.56 ± 0.44	0.49 ± 0.42	0.83 ± 0.11	**0.78 ± 0.14**

Table 5: T-Net performs better than baselines on ASW-Synth. Similar to 4, we report 5-fold cross-validation F_β results on ASW-Synth dataset for $\beta = 0.5, 1$. This dataset contains 1005 HT, 4794 ISW and 1437 Spam clusters.

descriptors such as age, ethnicity, etc., and does not use automatic text encoding approaches. This suggests there is a very low risk of encoded biases impacting the predictions. The clusters that get flagged as HT by T-Net can act as potential leads for investigators. Instead of scouring through large volumes of ads, they can start with investigating the suspicious clusters and easily find connected ads which also helps with case building. Additionally, identifying and filtering out spam can help remove noise from these datasets, and thus improve future studies in this domain.

HT detection is a complex problem that requires multifaceted solutions ranging from technical to social, policy and legal domains. T-Net will be integrated into a technical solution that is currently under development as a part of our broader project on countering human trafficking. T-Net will also be shared with our collaborators that develop tools to assist HT investigations, with potential for being included in their pipeline. The release of the synthetic dataset ASW-Synth and code base for T-Net including LFs for weak labeling have more immediate impacts on the research community by enabling more researchers to work on this problem.

Future works Our proposed methodology detects clustered activity which is easier to interpret than looking at ads independently. An interesting follow up here is to investigate the explainability aspects of the method and develop visualizations to interpret the results of T-Net. The labeling functions in T-Net are designed based on the current literature, and our consultations with domain experts and survivor leaders. These can be updated as the trafficking patterns change in the future to make sure the model stays robust and accurate. As the laws and policies in this domain are catching up with the technologies, we are now conducting research in close collaboration with domain experts, to determine the most ethical processes of using AI in this domain. The output of our algorithms should be used in practice considering the legal (privacy and AI laws), ethical (potential harms to vulnerable groups such as migrants and indigenous peoples) and human rights (privacy, surveillance, and the use of technology in criminal justice) considerations. There is also scope for improving the synthetic data generator such as by incorporating more nuanced prompts for making more realistic ads. Strategies for evaluating these generated ads also need to be studied.

Ethics Statement

The data we use is publicly available and due to the nature of the ads, there is no reasonable expectations for privacy. However, due to sensitivity of the data, an Ethics Approval has been obtained from the Research Ethics Board Office at the authors' university for using this type of data. We have also studied the current best practices for the project through a commissioned Responsible AI Institute evaluation, one of whose recommendation was to focus on a human centered design. To this end, we have biweekly consultations with human trafficking survivors and have been mindful of not reproducing biases in the design of the algorithm. No personal attributes such as age, physical descriptors, ethnicity, etc were used. Furthermore, the analysis done in this study is not on the individual level but more so on a cluster level where the focus is on narrowing down types of activity. Individual sex workers' ability to advertise their work online without being criminalized must not be jeopardized by this research. In addition, we have also reviewed the current law and policy implications through a comprehensive legal risk assessment and mitigation memorandum from a law firm. Lastly, for transparency and accountability, all of the algorithms being developed will be made available online and to maintain confidentiality and anonymity of the data, data scraping scripts and real-world datasets will not be shared publicly. The synthetically generated dataset will be made available on request via email to the corresponding author.

Acknowledgements

This research is partially funded by the Canada CIFAR AI Chairs Program and the first author is supported by the FRQNT scholarship. The project also received support from the National Science Foundation Graduate Research Fellowship under Grant No. DGE1745016 and DGE2140739, as well as the Pittsburgh Infrastructure Technology Alliance (PITA).

References

Alvari, H.; Shakarian, P.; and Snyder, J. K. 2017. Semi-supervised learning for detecting human trafficking. *Security Informatics*, 6(1): 1.

Arpit, D.; Jastrzębski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M. S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; et al. 2017. A closer look at memorization in deep networks. In *International conference on machine learning*, 233–242. PMLR.

Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10): P10008.

Canada, P. S. 2023. About human trafficking.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Crotty, S. M.; and Bouché, V. 2018. The red-light network: Exploring the locational strategies of illicit massage businesses in Houston, Texas. *Papers in Applied Geography*, 4(2): 205–227.

Dai, E.; Aggarwal, C.; and Wang, S. 2021. Nrgnn: Learning a label noise resistant graph neural network on sparsely and noisily labeled graphs. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 227–236.

Du, X.; Bian, T.; Rong, Y.; Han, B.; Liu, T.; Xu, T.; Huang, W.; Li, Y.; and Huang, J. 2023. Noise-robust Graph Learning by Estimating and Leveraging Pairwise Interactions. *Transactions on Machine Learning Research*.

Dubrawski, A.; Miller, K.; Barnes, M.; Boecking, B.; and Kennedy, E. 2015. Leveraging publicly available data to discern patterns of human-trafficking activity. *Journal of Human Trafficking*, 1(1): 65–85.

Durisin, E. M.; and van der Meulen, E. 2021. Sexualized Nationalism and Federal Human Trafficking Consultations: Shifting Discourses on Sex Trafficking in Canada. *Journal of Human Trafficking*, 7(4): 454–475.

Hassani, K.; and Khasahmadi, A. H. 2020. Contrastive multi-view representation learning on graphs. In *International conference on machine learning*, 4116–4126. PMLR.

Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Kulshrestha, A. 2021. *Detection of Organized Activity in Online Escort Advertisements*. McGill University (Canada).

Lee, M.-C.; Vajiac, C.; Kulshrestha, A.; Levy, S.; Park, N.; Jones, C.; Rabbany, R.; and Faloutsos, C. 2021. INFOSHIELD: Generalizable Information-Theoretic Human-Trafficking Detection. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, 1116–1127.

Li, Y.; Nair, P.; Pelrine, K.; and Rabbany, R. 2022. Extracting Person Names from User Generated Text: Named-Entity Recognition for Combating Human Trafficking. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2854–2868.

L'Hoiry, X.; Moretti, A.; and Antonopoulos, G. A. 2021. Identifying sex trafficking in Adult Services Websites: An exploratory study with a British police force. *Trends in Organized Crime*, 1–22.

Nagpal, C.; Miller, K.; Boecking, B.; and Dubrawski, A. 2017. An Entity Resolution Approach to Isolate Instances of Human Trafficking Online. In *Proceedings of the 3rd Workshop on Noisy User-generated Text, NUT@EMNLP*, 77–84.

Nair, P.; Li, Y.; Vajiac, C.; Olligschlaeger, A.; Lee, M.-C.; Park, N.; Chau, D. H.; Faloutsos, C.; and Rabbany, R. 2022. VisPaD: Visualization and Pattern Discovery for Fighting Human Trafficking. In *Companion Proceedings of the Web Conference 2022*, WWW '22, 273–277. New York, NY, USA: Association for Computing Machinery. ISBN 9781450391306.

Organization, I. L. 2022. Global estimates of modern slavery: forced labour and forced marriage. https://www.ilo.org/wcmsp5/groups/public/---ed_norm/---ipec/documents/publication/wcms_854733.pdf. Accessed: 2024-02-09.

Park, N.; Rossi, R.; Koh, E.; Burhanuddin, I. A.; Kim, S.; Du, F.; Ahmed, N.; and Faloutsos, C. 2022. Cgc: Contrastive graph clustering forcommunity detection and tracking. In *Proceedings of the ACM Web Conference 2022*, 1115–1126.

Portnoff, R. S.; Huang, D. Y.; Doerfler, P.; Afroz, S.; and McCoy, D. 2017. Backpage and Bitcoin: Uncovering Human Traffickers. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Project, P. 2014. National Human Trafficking Hotline. https://polarisproject.org/. Accessed: 2024-02-09.

Ratner, A.; Hancock, B.; Dunnmon, J.; Sala, F.; Pandey, S.; and Ré, C. 2019. Training complex models with multi-task weak supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 4763–4771.

Ratner, A. J.; De Sa, C. M.; Wu, S.; Selsam, D.; and Ré, C. 2016. Data programming: Creating large training sets, quickly. *Advances in neural information processing systems*, 29.

Reed, S.; Lee, H.; Anguelov, D.; Szegedy, C.; Erhan, D.; and Rabinovich, A. 2014. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*.

Stylianou, A.; Souvenir, R.; and Pless, R. 2019. Traffick-Cam: Explainable Image Matching For Sex Trafficking Investigations. *arXiv preprint arXiv:1910.03455*.

Szekely, P.; Knoblock, C. A.; Slepicka, J.; Philpot, A.; Singh, A.; Yin, C.; Kapoor, D.; Natarajan, P.; Marcu, D.; Knight, K.; et al. 2015. Building and using a knowledge graph to combat human trafficking. In *International Semantic Web Conference*, 205–221. Springer.

Thorn, D. V. B. 2015. A Report on the Use of Technology to Recruit, Groom and Sell Domenstic Minor Sex Trafficking Victims. https://www.thorn.org/wp-content/uploads/2015/02/Survivor_Survey_r5.pdf. Accessed: 2024-02-09.

Tong, E.; Zadeh, A.; Jones, C.; and Morency, L.-P. 2017. Combating human trafficking with deep multimodal models. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL*, 1547–1556.

U.N. 2021. U.N News. Traffickers abusing online technology, UN crime prevention agency warns. https://news.un.org/en/story/2021/10/1104392.

Vajiac, C.; Chau, D. H.; Olligschlaeger, A.; Mackenzie, R.; Nair, P.; Lee, M.-C.; Li, Y.; Park, N.; Rabbany, R.; and Faloutsos, C. 2022. TRAFFICVIS: visualizing organized activity and spatio-temporal patterns for detecting and labeling human trafficking. *IEEE transactions on visualization and computer graphics*.

Van Rooyen, B.; and Williamson, R. C. 2017. A Theory of Learning with Corrupted Labels. *J. Mach. Learn. Res.*, 18(1): 8501–8550.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33: 5812–5823.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2020. Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*.