# Active Link Inference for Case Building Investigation


## Abstract

How can we infer connections between entities, given the evidences of their relations? How can we help an investigator to find salient patterns by connecting the dots more efficiently? For example in combating online human trafficking, connecting the related escort advertisements outlines an organized activity, which is the first step to identify the target who is advertising the victims. This paper presents a principled way of inferring links where the user is able to actively provide feedback and guide the inference. We show how our method helps an investigator to find related entities to an initial lead and explains why they are related, when mining millions of escort advertisements posted in one of the largest classified advertising websites. As an active link inference method, our proposal has broad applications wherever the connections between entities are not given a priori, in particular for mapping out the covert web of entities in fraud detection and counterterrorism.


## 1 Introduction and Related Works

In many applications, we are interested in how different datapoints are related to each-other, in addition to what are the characteristics of each individual datapoint. In most cases, the relation between the datapoints should be inferred from the available data. We are proposing an active network inference algorithm for such settings which supports large scale data while incorporating the user's feedback to guide the network inference. We are tailoring this approach for combating human trafficking in online escort advertisements to find how different advertisements are linked together and point to organized activities; e.g. advertisements for different potential victims which are linked by phone numbers, catch phrases or text patterns, images with the same background, or other evidences of connection. For case building and target identification in this domain, an initial lead is treated as the seed query to find connected entities and identify the person of interest. Currently, this task is performed by an expert investigator through manual exploration of the available data. Our proposed algorithm makes this process semi-automatic and more efficient.

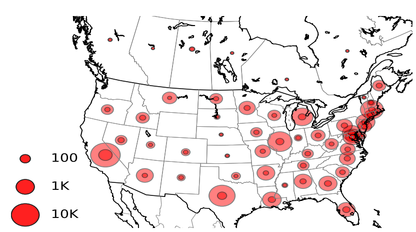


Figure 1: Reported cases to a human trafficking hotline. This maps shows the distribution of online escort advertisements in our data which use a phone number associated with the reported human trafficking cases [6].

More precisely, our data includes a sample of millions of escort advertisements scraped from Backpage.com for cities across the US and Canada posted between August 2013 to January 2017 obtained from [2]. We are modelling this data as a k-partite graph, in which advertisements are connected to different types of evidences they share, e.g. phone numbers, bigrams, images, etc. The user's feedback is then used to guide the navigation through this graph when finding related advertisements. Here, we adjust the weight and importance of each of the evidence modalities (e.g. phone numbers would become stronger indicators than bigrams), as well as the relevance of specific evidences (e.g. some phone numbers are relevant to the current seed). The main intuition of the proposed approach is that the candidates are explored in order of how well connected they are to the positive nodes already explored (weighted sum of all paths which shows the shared evidences) whereas importance of connections (weights) are learned based on user's

Submitted to Advances in Modeling and Learning Interactions from Complex Data (AMLICD 2017). Do not distribute.

feedback. Our proposed method is compared against different baselines, including a random-walk with restart at negative feedbacks.

Our method uses selective labeling as in active learning which enables analyzing data where labels are scarce. In this setting, the task of recovering only the relevant portion of the data, is referred to as active search [4]. Active search is similar to active binary classification, however the objective is to achieve the highest recall for a given class of interest, instead of the overall classification accuracy. When targeting a specific class is the goal, e.g. for detecting fraud, or drug discovery, the active search technique achieves better performance compared to the uncertainty sampling common in active classification [8]. Here, the utility function is defined to be the exact recall objective which is then optimized given the fixed query budget. More relevant to our work is active search on graphs formulated in [8], where given the relations between the entities, the goal is to find the entities of the same label. In our case, the relationships are not given a priori and should be inferred based on the user's feedback. Note that we are interested in explainable connections backed by shared pieces of information, not mere similarity. A basic assumption in the active search is that the target class clusters together based on some similarity measure [4]. In this sense, the problem closely resembles active local clustering which is even more relevant to our work, as it looks for only positive entities which are connected by the given evidences, i.e. are explainable. Clustering nodes in graphs is a widely studied problem, a.k.a. community detection. Here, we are interested in a subclass of local clustering algorithms, which are generally applied to tackle the volume of the large scale data. As oppose to the global clustering algorithms that partition a given graph, the local methods retrieve a cluster by expanding from a given seed node. Having such an algorithm, one can take a peeling strategy to cluster all the network – detecting and removing clusters one by one, e.g. see [1]. From the many algorithms proposed for local clustering in graphs, the ones that consider attributes or labels for nodes and cluster heterogeneous graphs are more relevant to our work [1, 7]. These approaches consider labels or attributes as an additional information source or meta data on nodes, and develop unsupervised algorithms to cluster the data. We are proposing a semi-supervised or active paradigm, since labels are not readily available, and we are acquiring the labels provided by the expert user while retrieving relevant entities. Moreover, although the relevant entities are assumed to be highly connected, our objective is different that the clustering algorithms as we are interested to find highly connected entities with positive labels which reflect the user's interest.

## 2 Methodology

Consider $n$ datapoints $\mathcal{D} = \{d_1, d_2 \dots d_n\}$ connected to $k$ different types of evidences (e.g. phone number, image, bi-grams) which we refer to as modalities. Let evidence set $\mathcal{E} = \{\mathbf{X}_1, \mathbf{X}_2 \dots \mathbf{X}_k\}$ denote the set of indicator matrices for these $k$ modalities, i.e. $\mathbf{X}_m \in \mathbb{R}_+^{n \times c_m}$ for $m \in [1 \dots k]$ where $c_m$ is the cardinality (number of unique evidences) of modality $m$ (e.g. number of unique phone numbers). Each column of $\mathbf{X}_m$ shows a set of datapoints that share the corresponding evidence (e.g. datapoints that all share a particular phone number), and each row of $\mathbf{X}_m$ shows the evidences associated to the corresponding datapoint (all the phone numbers mentioned in a particular datapoint). Given this data, we consider:

**Definition 1 (Active Link Inference)** *Given seed datapoint of interest, $i$, and assuming an unknown label set $\mathbf{y} = \{y_1, y_2, \dots, y_n\}$ where $y_j = 1$ if the user deems node $y_j$ related to the node $i$ and zero otherwise, find the maximum number of related entities to $i$ given a fixed query budget, $b > 0$.*

Considering each indicator matrix shows the biadjacency matrix of a 2-partite graph, we build a k-partite graph representation for the data. Given which and seed node of interest $v_i$, the Active Link Inference translates to finding positive nodes that are (tightly) connected to node $v_i$ with length even paths (i.e. through shared evidences); whereas positive nodes means when we query label of the found endpoint $v_j$, we have $y_j > 0$. Our proposed method navigates through this graph to efficiently find these nodes while learning the importance of each modality and each piece of evidence from the labels (user's feedbacks) obtained while expanding.

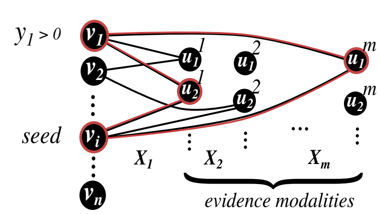


Figure 2: evidence graph.

We consider a weight for each partition of the evidence graph to enforce the importance of its corresponding modality in the given evidence set, i.e. $\mathbf{\Theta} = [\theta_1, \theta_2, \dots \theta_k]$; $\theta_m$ denotes the weight/importance of modality $m$. Correspondingly, we consider the evidence flow coming from each partition separately, i.e. considering a tie-strength vector for each expansion candidate $v_j$ as: $\mathbf{s}^j = [s^j_1, s^j_2, \dots s^j_k]$ where $s^j_m$ shows the tie strength of reaching $v_j$ from evidences in modality $m$. We can consider $s^j_m$ simply as the number of (length two) paths that go through evidences in partition $m$ to reach $v_j$. To enforce the hypothesis that rare evidences are more important, we further weigh down the evidences by their prevalence (the number of datapoints they are associated with) measured as their degree in the graph. In more detail, if node $v_i$ and $v_j$ are connected through evidence $u$, this evidence contributes to their tie strength by $1/d_u^2$, as each edge is down weighted by the degree. Moreover, the tie strength score of reaching $v_j$ is summed from all the currently explored (labeled) positive nodes, $L^+ = \{j \in L | \; y_j > 0\}$, i.e.

$$s^j_m = \sum_{v_i \in L^+} \sum_{u \in \mathcal{N}(v_i) \cap \mathcal{N}(v_j)} \frac{1}{d_u^2} \tag{1}$$

To infer the next node, we pick $j^* \leftarrow \arg\max_j \sum_m s^j_m \theta^j_m$ as the node with highest overall evidence support. Now let node $v_j$ denote this last queried node, for which we observe the label $y_j$. We adjust the modality which most supported the selection of node $j$, i.e. we first determine the support modality, $m^* = \arg\max_m s^j_m \theta_m$; then we adjust the importance/credit of the modality $m^*$ as $\theta^*_m = \delta \theta^*_m$ if $y_j < 0$ and $\theta^*_m = (2 - \delta)\theta^*_m$ if $y_j > 0$; where $\delta \in (0, 1)$ is the learning rate. Given the observed labels, one can also adjust evidence flow weights, assuming some pieces of evidence (within or across different modalities) are more relevant to the seed. To enforce this, we also consider the negative flow that passes through an evidence. In more detail, instead of Equation 1 we use:

$$s^j_m = \sum_{v_i \in L^+} \sum_{u \in \mathcal{N}(v_i) \cap \mathcal{N}(v_j)} \frac{1}{d_u^2} - \sum_{v_i \in L^-} \sum_{u \in \mathcal{N}(v_i) \cap \mathcal{N}(v_j)} \frac{1}{d_u^2} \tag{2}$$

where $L^- = \{j \in L | \; y_j < 0\}$.

To evaluate the performance of the the proposed method and the effect of different its components, we consider four baselines described as follow.

**No Feedback (NF)**: baseline uses the scoring scheme described above to expands from the node with the current maximum score, but ignores the feedback from the user. The resulted algorithm is similar to a local clustering where expansion is purely based on the connectivity.

**No Modality (NM)** baseline uses the feedback to re-weigh the evidences; however the importance of different modalities is not adjusted based on the feedback.

**No Negative Evidence Flow (NE)** baseline uses the feedback to learn the importance of different modalities but ignores the negative instances.

**Random Walk (RW)** baseline expands from the seed by randomly walking through its neighbors. This walk restarts from the seed whenever it reaches a negative node. In this way, the random walk is taking into account the feedbacks by only expanding on positives. We are however not learning from what caused the algorithm to reach to a positive or negative instance.

# 3 Experiments

For each advertisement, we have access to its unstructured text (title and body), attached images, date and location posted. We use a publicly available regular expression extractor, which is developed for the same source of data previously[3] to extract basic features from the advertisement text, which are: phone number, email, url, and name. We also consider unigrams and bigrams used in both title, and body.[1]. From these advertisements, we build two datasets, one of roughly 2.5 million ads posted in DC, Maryland, Virginia area (HT-DMV), and another one of about 4 million ads posted between July to December 2013 (HT-13), which is chosen to include activities associated with a list of phone numbers reported to a victim advocacy groups [3], the spatial frequency of which was plotted earlier in Figure 1. For the evaluation of the algorithms, and since labels are not available in these datasets, we keep some modalities that strongly indicate relations between ads to derive labels. In more detail, we construct labels by connected components formed when only using url, email, and phone numbers. This means that two ads are assumed 'truly' related only if they share either of these hard identifiers.

We also include Discogs from KONECT[5], a publicly available datasets in our experiments. This dataset is collected from a large online music database, and provides information about 3.5 million

[1] filtering those that appear in more than 10k advertisements, to trim out stop word and common phrases.

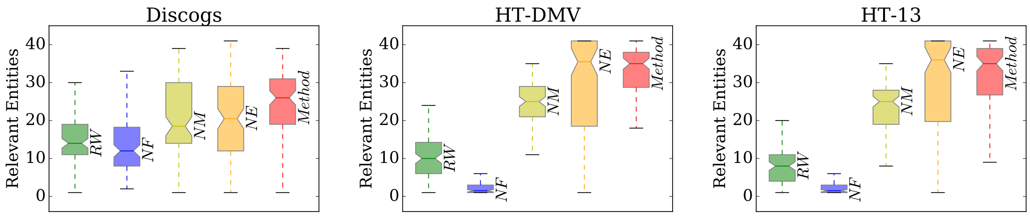


Figure 3: Comparison with baselines on three datasets.

different releases: date of the release, artists (primary and extra) involved, record labels, track information, companies involved in the production of the release, etc. In this dataset we treat artists as true labels and consequently we are learning to find releases of the same artist as the given seed release based on their shared i

Figure 3 shows the number of relevant entities found by the proposed method and the four baselines, on three datasets, in terms of how many relevant entities they found within a fixed query budget of 40. We can see that indeed expanding only based on topology and ignoring the user's feedback (NF) is showing the poorest performance. Moreover, the random walk (RW) method which uses the feedback to restart but ignores the importance of different modalities and doesn't remember the factors that resulted in reaching a negative, is only doing slightly better. The variations which only consider weights on evidences (NM) or modalities (NE) are also not doing as good as when incorporating both as in the proposed method.

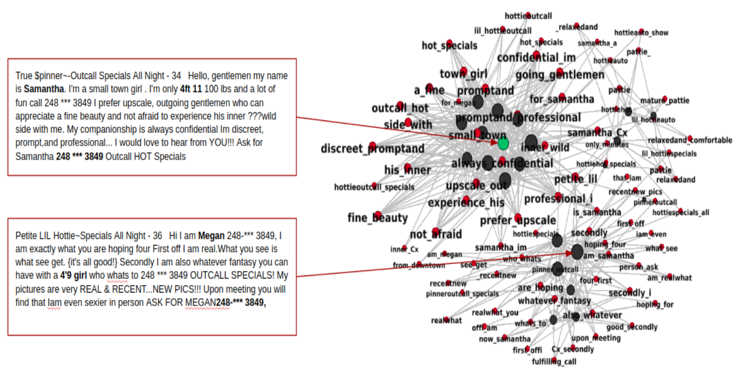


Figure 4: Green advertisement shows the seed advertisement used in this case study. The black nodes are the related advertisements discovered which are connected to the seed though shared evidences, plotted as red nodes. Here near duplicates (repeats over time) and high confidence matches (possible repeats) are filtered out. We can see the original text used in the seed advertisement in the box on the top left. The bottom left box shows the text of a connected advertisement which is advertising a different person. Although these two text don't show high similarity, these two ads are truly related as they share the same phone number. The ANI was able to correctly discover it, and this representation explains how these ads are connected through shared evidences.

Here, we use a general purpose graph visualization tool for plotting example results. However, a specific graphic user interface should be developed to facilitate deriving and navigating through the results.

## References

[1] N. Ailon, Y. Chen, and H. Xu. Iterative and active graph clustering using trace norm minimization without cluster size constraints. *Journal of Machine Learning Research*, 16:455–490, 2015.

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

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

[4] R. Garnett, Y. Krishnamurthy, D. Wang, J. Schneider, and R. Mann. Bayesian optimal active search on graphs. In *Ninth Workshop on Mining and Learning with Graphs*, 2011.

[5] The koblenz network collection. `http://konect.uni-koblenz.de/`, 2017.

[6] Polaris project. https://polarisproject.org/, 2014.

[7] B. Perozzi, L. Akoglu, P. I. Sánchez, and E. Müller. Focused clustering and outlier detection in large attributed graphs. In *KDD*, pages 1346–1355. ACM, 2014.

[8] X. Wang, R. Garnett, and J. G. Schneider. Active search on graphs. In *KDD*, pages 731–738. ACM, 2013.