Structural Trend Analysis for Online Social Networks

Ceren Budak
Department of Computer Science, UCSB
Santa Barbara, USA
cbudak@cs.ucsb.edu

Divyakant Agrawal
Department of Computer Science, UCSB
Santa Barbara, USA
agrawal@cs.ucsb.edu

Amr El Abbadi
Department of Computer Science, UCSB
Santa Barbara, USA
amr@cs.ucsb.edu

ABSTRACT

The notion of trends in social networks has emerged as an important problem attracting the attention of researchers as well as the industry. Although, recent work has studied trends from various perspectives such as its temporal and geospatial properties, the structural properties of the network that creates such trends are ignored in trend detection. In this work, we propose two novel structural trend definitions called *correlated* and *uncorrelated* trends that leverage friendship information to detect interesting topics that would not be detected using traditional trend definitions. We experimentally and analytically show that *correlated* trends are significantly different from *traditional* trends whereas the difference for *uncorrelated* trends, although corresponding to a useful variation, is less pronounced. We show that both *correlated* and *uncorrelated* trends identify interesting activities in social networks. We also show that the new trend definitions can be used to detect or filter suspicious activity in the network. Detection of structural trends is inherently harder than traditional trend detection. Therefore we propose a sampling technique that provides computational gain while remaining within an acceptable error bound. Experiments performed on a large-scale social network data of 41.7 million nodes and 417 million posts show that even with a small sampling rate of 0.005, the average precision lies above 0.93 for *correlated* trends while keeping a perfect average precision of 1 for *uncorrelated* trends.

1. INTRODUCTION

Social networks provide large-scale information infrastructures for people to discuss and exchange ideas about different topics. Detecting trends of such topics is of significant interest for many reasons. For one, it can be used to detect emergent or suspicious behavior in the network. They can also be viewed as a reflection of societal concerns or even as a consensus of collective decision making. Understanding how a community *decides* that a topic is trendy can help us better understand how ad-hoc communities are formed and how decisions are made in such communities. In general, constructing "useful" trend definitions and providing scalable solutions for them will contribute towards a better understanding of human interactions in the context of social media.

Trends in social networks have recently been a major focus of interest among researchers studying them from perspectives such as temporal [24] and geographical dimensions [32, 30]. A similar interest can be observed in industry. For instance, Twitter trends [37] have been a testament to societal concerns, to such an extent that when there was interest in Wikileaks and the hashtag #wikileaks did not appear in the trends list in Twitter, there was substantial discussion upon which Twitter had to make an official announcement stating they have excluded #wikileaks from the trending topics [40]. The large number of companies reporting trends in Twitter is another testament to the importance of trends [37, 18, 19].

Although trends in social networks have been extensively studied, to our knowledge all the published work in this area ignores the structural properties of the social network that created these trends. In today's social networks where users are highly influenced by their friends, trend definitions that reach beyond simple heavy-hitters approaches to integrate the importance of such flow of influence can be of great benefit. The main purpose of this paper is to define two such trend definitions, emphasize their significance and provide efficient online solutions for them. Since information diffusion on a social network is a substantial part of the process that creates the information trends, properties that are defined in this context are of significant interest. For example, consider a group of friends in a large social network like Facebook discussing an attack. Detecting this new interest of this specific group on "attacks" can be of great importance. We aim to address this problem using structural trends. In essence, a structural trend is a topic that is "hot" within structural subgroups of the network. The challenges are to formally define the notions of a structural subgroup and to develop techniques to detect *structural trends*.

As a starting point, we consider the problem of identifying the number of connected pairs of users in a social network that are discussing a specific topic. We refer to this as detecting *correlated trends*. This trendiness definition will bias topics that are discussed among clustered nodes in the network. Alternatively, one might be interested in the number of *unrelated* people interested in a specific topic and in trends that results from this interest. We call these *uncorrelated trends*. This definition of trendiness can be used to capture the notion of the *trustworthiness* of a trend. In this case the trendiness of a topic will not be biased by a discussion amongst a small clustered group. Large-scale online social networks such as Twitter and Facebook provide tools for millions of people to share information. Although current trend definitions used by the industry such as trending topics of Twitter are good at detecting trends at global scale, their shortcomings such as their vulnerability to spammers or inability to detect interesting activity in different communities make them less valuable from an analytical perspective [38]. The new trend definitions introduced in this paper provide methods for a deeper analysis of activity in social networks. We will demonstrate the value of *structural* trends by identifying the types of topics detected using this new definitions that would otherwise be undetected. Although, structural trends enable a deeper analysis of information shared in social networks, their detection is harder to identify than the traditional heavy hitter-based definition of trends since the counting scheme needed for *structural* trends calls for new, graph-oriented solutions. Considering both the large scale of social networks as well as the sheer volume of information shared, we introduce a sampling based technique that provides efficiency while still remaining within an acceptable error bound.

To our knowledge, this is the first work that incorporates the structure of a graph and the connections between agents that create trends to the definition of trends. We introduce two new definitions of trendiness based on the structure of the network and study the significance of these new definitions. We experimentally and analytically show that *correlated* trends are significantly different from *traditional* trends whereas *uncorrelated* trends tend to be more similar to *traditional* trends while filtering out "spammy" topics from trends. We show that *structural* trends identify interesting activities in social networks. In Section 2 we will first start with a brief overview of related work. In Section 3, we will formally define the notions of *correlated* and *uncorrelated* trend. Later in Section 4 we will demonstrate the significance of these definitions by identifying the types of topics they detect as structurally significant. Section 5 provides sampling-based solutions for *correlated* and *uncorrelated* trends. In Section 5, we experimentally study the accuracy and efficiency of the solutions provided which show that a high accuracy of over 0.93 can be achieved even with a small sampling probability of 0.005. Finally Section 6 concludes the paper.

2. RELATED WORK

Trends in social networks have recently been a focus of interest for many researchers. Kwak et al. [23] study information spread on Twitter to detect trending topics and compare trending topics in Twitter with those in other media, namely, Google Trends and CNN headlines. The results show that the majority (over 85%) of topics are headline news or persistent news in nature. The way they identify trends is substantially different than what we propose in this paper. They use the traditional counting mechanism on time slices by considering a trending topic inactive if there is no tweet on the topic for 24 hours. Leskovec et al. [24] also study temporal properties of information shared in social networks using blogosphere data. They focus on tracking new topics, ideas, and "memes" across the web and studying their temporal properties by developing scalable algorithms for clustering textual variants of "memes". Some other works that identify topics over time are [1, 16].

Another important characteristic of news or discussions in social networks is the spatial properties of the agents that are involved in the discussion or the source of the news. A recent work by Teitler et al. [32] collects, analyzes, and displays news stories on a map interface, thus leveraging their implicit geographic context. A follow-up study performs similar techniques to identify geographical information in news in Twitter [30]. Although these works that focus on temporal and spatial characteristics of trends are important for a better understanding of the notion of trends, they are orthogonal to the approaches introduced in this study. Unlike earlier studies, we focus on structural properties of the network that create trends.

Studying trends from a structural point of view requires using efficient solutions that can handle the large scale of the social networks. More often than not, this requires developing approximation algorithms. Since trends are time-sensitive, offline solutions that require a non-constant number of passes on data are impractical. In this setting, one needs to employ some sort of a streaming solution. A simple definition of trendy topics can be the frequent items throughout the entire stream of user activities. The problem, defined this way, is simply to find the frequent items in a stream of data, also referred to as *heavy hitters*. The *frequent elements problem* is well studied and several scalable, online solutions have been proposed [11, 12, 28, 25, 13]. Unlike solutions based on such techniques, the solution provided in this work is not oblivious to the graph structure that creates the stream.

Lately, a number of works have studied structural properties of graphs in a streaming or semi-streaming fashion. A type of problem that is significantly related to the problem studied in this paper is counting triangles in a graph stream. There are three types of solutions to this problem: exact counting [3], streaming [4, 9, 21] and semi-streaming algorithms [5, 35]. Although streaming algorithms [4, 9, 21] provide efficient solutions, they solve the global triangle counting problem, which counts *all* the triangles in a graph whereas structural trendiness requires solutions closer to local triangle counting (i.e., compute score per topic rather than score of all topics combined). In that sense, problems studied in [4, 9, 21] are closer to the problem studied in this paper. However, these works provide a semi-streaming solution. Detecting trends requires online solutions and therefore such techniques are not applicable.

3. PROBLEM DEFINITION

L Consider a directed graph $G = (N, E)$ representing a social network consisting of nodes N and edges E. A node n_i is a neighbor of n_j if and only if there is an edge $e_{j,i}$ from n_j to n_i in E. Nodes can choose to (or not) share a certain piece of information which is subsequently visible to their *neighbors*. Each such information belongs to one or more *topics* which can be identified. Therefore, each piece of information shared by node n_i on a specific topic T_x can be modeled as a tuple $\langle n_i, T_x \rangle$. Note that, topic extraction is a hard problem in its own right and we will not be focusing on this problem in this paper.

In order to identify trendiness of a topic T_x in a conventional manner, one could count the total number of times T_x is discussed. Namely. *traditional* trendiness of a topic T_x can be computed as:

$$f(T_x) = \sum_{n_i \in N} C_{i,x} \tag{1}$$

where $C_{i,x}$ represents the number of tuples of the form $\langle n_i, T_x \rangle$. This trend definition is completely oblivious to the structure of the network graph. We propose two new alternative trend definitions, namely *correlated* and *uncorrelated* trends, to capture trending topics of different nature that incorporate the network structure. The trendiness of a topic T_x using *correlated trends* definition captures the number of connected pairs of nodes in graph G talking about T_x. Equation 2 provides the value (or score) of a specific topic under this new trend paradigm. The score function defined in Equation 2 is meant to capture *all possible forms of influence propagation between any two neighboring nodes*. Consider a stream of broadcasts: ... b_1:$\langle n_1, T_x \rangle$,...,b_2:$\langle n_1, T_x \rangle$, b_3:$\langle n_2, T_x \rangle$,...,b_4:$\langle n_2, T_x \rangle$,...b_5:$\langle n_1, T_x \rangle$... where n_1 and n_2 are neighbors and b_i are a subset of broadcasts in the broadcast stream. In this setting b_1 and b_2 *might have influenced* node n_2 to share b_3 and b_4 broadcasts since b_1 and b_2 precede b_3 and b_4. Similarly b_3 and b_4 might have influenced n_1 to share b_5. All the pairs of *possible flow of influence* sum up to 6 in this example. In general, Equation 2 captures this characteristic for arbitrary topics and arbitrary undirected graphs. For directed graphs, although *correlated trend score* does not correspond directly to this notion, it can still be seen as capturing a similar behavior. Also, with a small change to the score definition to count pairs only in

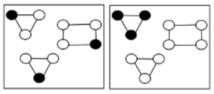


Figure 1: Black nodes represent nodes talking about topic T_x, whereas white nodes represent the nodes that are not

forward time can capture the same behavior for directed graphs. The proofs of approximation of the sampling method can be shown to hold for that case as well, though the explanation of approximation bounds is more complicated for that setting. For simplicity, Equation 2 simply counts pairs of nodes discussing a topic.

$$g(T_x) = \sum_{n_i \in N} (C_{i,x} \sum_{n_k \in N_i} C_{k,x}) \tag{2}$$

where $N_i = \{n_k | e_{i,k} \in E\}$. This function assigns high scores to topics that are discussed heavily in a cluster of tightly connected nodes. Consider the two graphs in Figure 1. The black nodes correspond to people that are talking about a specific topic T_x and white nodes are people who are not talking about T_x. Even though both graphs have the same number of people talking about T_x, in the graph on the right, the people talking about T_x are a part of a more clustered subgraph, giving the topic a higher structural significance. $g(T_x) = 6$ for the graph on the right whereas, $g(T_x) = 0$ for the graph on the left as there are no connected pairs talking about it.

In comparison, *uncorrelated trends* aim to capture the behavior at the other extreme, i.e., where we are interested in the number of *unrelated* people interested in a specific topic and in trends that results from these unrelated people:

$$h(T_x) = \sum_{n_i \in N} (C_{i,x} \sum_{n_k \in (N - n_i - N_i)} C_{k,x}) \tag{3}$$

This definition favors topics that a large number of unrelated people are interested in. Going back to our example of two graphs in Figure 1, for the graph on the left $h(T_x) = 6$, whereas $h(T_x) = 0$ for the graph on the right. As discussed before, this definition of trendiness can be used to capture the notion of the *trustworthiness* of a trend. In this case the trendiness of a topic is not biased by a discussion in a small clustered group.

We denote *top-k* topics w.r.t. their f, g, h scores as *traditional, correlated* and *uncorrelated* trends respectively. We refer to the combined class of *correlated* and *uncorrelated* trends as *structural trends*. In the following sections, we will demonstrate the usefulness of *structural* trends and provide solutions for detecting them.

4. STRUCTURAL TRENDS SIGNIFICANCE

In this section, we will demonstrate the value proposition of structural trends by identifying the "interesting activity" automatically detected using such new trend definitions. We will demonstrate the significance of the structural trends defined in Equations 2 and 3 by addressing the following questions: 1) Are the *structural* trends different from *traditional* trends? 2) What is the nature of structural trends? Are there interesting characteristics that can be identified using parameters of the network or the information diffusion process?

We make use of two different methods to answer these questions. First, we develop a model of diffusion of an arbitrary number of information campaigns in a social network. The importance of structural trends is then identified with respect to the parameters of this model. Second, we analyze data from Twitter, a large-scale online social network and identify the types of topics identified using structural trends and focus on their significance.

4.1 Model-Based Value Proposition

In order to systematically evaluate the significance of *structural trends*, we need to identify characteristics of social networks or topics that validate the value proposition of such definitions. To this end, we need to model the process that creates trends in a social network. Although there are a number of models of diffusion of one information campaign [22], there is little research on modeling of concurrent information campaigns with the exception of [8, 7, 10] which study diffusion of *two* concurrent campaigns. We introduce a natural extension of widely used Independent Cascade model [22] that models diffusion of an arbitrary number of campaigns. We call this model the *Independent Trend Formation Model (ITFM)* as the diffusion of topics are modeled to be independent of each other.

ITFM captures nodes as entities that are influenced by their neighbors as well as external entities such as news media. We model a social network as a directed graph. There are a set of m topics $T = \{T_1, ..., T_m\}$. Information diffusion proceeds in discrete time steps. At each step, nodes share information about zero or more topics with its friends. As we would like to model the different types of influence, we assign two types of probabilities to each node n_i: $p_{i,x}$ and $q_{i,j,x}$ that denote the probability that n_i will share information about topic T_x independently from any of its neighbors (external influence such as news media) and the probability that n_i will re-share some information about a topic T_x that its neighbor n_j shared in the earlier discrete time step (peer influence). If for a topic T_x the p probabilities are high, T_x spreads mostly through the news media channels, whereas if the q probabilities are high, this means T_x is *viral*, spreading through peer influence.

Our goal is to study the significance of *structural* trends using *ITFM*. To this end, we performed experiments on synthetic power-law graphs. Since social networks have power-law degree distribution [29], the synthetic data sets should have this property. There are various models of network formation that result in a power-law graph of degree distribution. We refer the reader to [29] for an extensive list. In this study we used the *Nearest Neighbor* model as it is shown to accurately capture various statistical metrics of real social network graphs [29]. There are two important parameters for Nearest Neighbor Model, u, i.e. the probability two nodes with a distance of two are connected at a time step and k, i.e. the number of pairs of existing nodes connected at a time step. We used $u = 0.8$ and $k = 1$ since it is stated in [29] these settings fit a real social network, namely Facebook Monterey Bay Network. The experiments explained in this section were all performed on a 500 node power-law graph with a set of 50 possible topics.

Question 1: Are traditional trends a good representative of structural trends? The first set of experiments were aimed at answering the question: Do structural trends provide extra information that could not be obtained otherwise? Or in other words, *how similar are structural and traditional trends?* To measure similarity we used *Spearman correlation coefficient* [27]:

$$\rho = 1 - \frac{6 \sum {d_x}^2}{n(n^2 - 1)} \tag{4}$$

where d_x is the difference between the ranks of topic T_x under the two trend definitions. *Spearman correlation* assesses how well the relationship between two variables can be described using a monotonic function. A perfect Spearman correlation of +1 (or -1) occurs when the variables are perfect monotonically increasing (or decreasing) functions of the other. We measured the *Spearman correlation coefficient* of *traditional* trends with *correlated* and *uncorrelated* trends using three experiments with different q settings (0.1, 0.3, 0.5), with all other variables fixed. Left two columns in Table 1 show that as the social network exhibits an increasingly viral behavior with increasing q values, both *correlated* and *uncorrelated* trends diverge from the traditional trends. The divergence is faster for *correlated* trends than that of *uncorrelated* trends.

Table 1: Model Similarity Statistics

q	$\rho_{trad-corr}$	$\rho_{trad-uncorr}$	AP_{corr}	AP_{uncorr}
0.1	0.762	0.988	0.140	0.569
0.3	0.640	0.976	0.095	0.466
0.5	0.600	0.965	0.083	0.398

Equation 4 represents how similar the rankings of *all* the topics are under the two trend definitions. However, in most cases the rankings of unpopular topics is of little significance. Our goal is to identify *structural* trends, i.e. top-k *correlated* ($top–k_{corr}$) and *uncorrelated* ($top–k_{uncorr}$) topics. Therefore it is more important to observe the similarity between $top–k_{corr}$ (or $top–k_{uncorr}$) and $top–k_{trad}$, i.e. *traditional* trends. In order to evaluate how good $top–k_{trad}$ topics are at mimicking or detecting $top–k_{corr}$ (or $top–k_{uncorr}$), we use *average precision*, an IR technique used to evaluate score of a ranked list of documents for a query. *Average precision* incorporates precision and recall values while evaluating a detection algorithm and can be computed as:

$$AP = \frac{\sum_{i=1}^{|D|} Prec(R_i)}{|D|} \tag{5}$$

where $D = \{d_1, d_2, ..., d_m\}$ is the set of relevant documents, R is the ranked set of documents retrieved by the detection algorithm and R_i is the set of ranked documents in R until document d_i is reached [26]. If d_i is not detected at all by the detection algorithm, $Prec(R_i) = 0$. We performed tests evaluating the *average precision* of $top–5_{trad}$ topics w.r.t. the relevant document set of $top–5_{corr}$ (or $top–5_{uncorr}$) topics. The results are given in Table 1 in columns AP_{corr} and AP_{uncorr} respectively and reflect similar results obtained using *Spearman correlation coefficient* on the entire topic list. Similar experiments where all parameters except p values are fixed reveal that with increasing p values, similarity between *traditional* and *uncorrelated* trends increases. This adheres to the intuition that, as p values dominate q values resulting in a setting where peer influence becomes less important, there is a smaller number of “spammy” topics for *uncorrelated* trends to filter out. For completeness purposes, we give the summary of the findings for this set of experiments in Section C in the Appendix.

Question 2: What is the nature of topics detected using structural trends? The second set of experiments evaluates the value of structural trends. As we demonstrated earlier, structural trends tend to be different from traditional trends. But what do such differences corresponds to? In the earlier set of experiments, the p and q values for each node and topic were chosen uniformly randomly from one distribution. Therefore, all the topics *had similar nature*. In the second set of experiments, half of the topics were chosen uniformly randomly from one distribution ($q = p = 0.1$) while the second set of topics are chosen from another distribution.

Consider a network G and a set of topics T. W.l.o.g. let T' denote the set of topics in T with $q' = 0.1$ and $p' = 0.1$, and the rest $T - T'$ topics be denoted T''. Setting $q < 0.1$ for a specific topic T_x would result in T_x spreading less significantly through social ties and setting $p > 0.1$ would balance this shortcoming by spreading through external influences (such as news media). Therefore, for the set T'', there can be another distribution that has $q'' < 0.1$ and $p'' > 0.1$ (or $q'' > 0.1$ and $p'' < 0.1$) values such that the average traditional ranking of T' and T'' are very similar. Next we test, how do topics in the subset T'' rank compared to T' w.r.t. their correlated and uncorrelated scores. Let us denote $AvgR_{trad}$, $AvgR_{corr}$ and $AvgR_{uncorr}$ of T' (or T'') as the average ranking of topics that belong to T' (or T'') w.r.t. the score function defined in Equation 1, 2 and 3 respectively. As the data set of this experiment consists of 50 topics, the topics rank from the highest score of 0 to 49. As could be expected, Table 2, shows that when $q'' < 0.1$ and $p'' > 0.1$, the *correlated* significance of topics in set T'' is much lower than T', while *uncorrelated* significance is higher. The opposite behavior is observed when the settings are $q'' > 0.1$ and $p'' < 0.1$. For instance, an average of 12 among 25 topics in T'' indicate that all the topics in T'' rank in top-25 *correlated* trends.

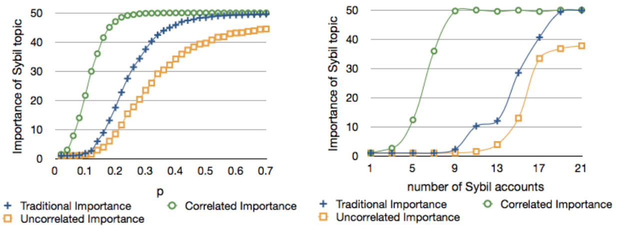
Table 2: Various Ranking Statistics

		$AvgR_{trad}$	$AvgR_{corr}$	$AvgR_{uncorr}$
$p' = 0.1, q' = 0.1$	T'	24.44	36.52	18.56
$p'' = 0.032, q'' = 0.15$	T''	24.56	12.48	30.44
$p' = 0.1, q' = 0.1$	T'	24.64	12	34.68
$p'' = 0.2, q'' = 0.054$	T''	24.36	37	14.32

Possible use case of structural trends: detecting or filtering Sybil activity: In a social network, in addition to topics having various characteristics, the nodes of the network can also vary w.r.t. their behaviors. A subset of the users can be interested in a set of topics while others are interested in different topics. Similarly, some nodes might be more influenced by their neighbors while others are more influenced by external entities such as news media. One usefulness of structural trends would be if malicious activity by a *subset of users* can be detected or filtered out using the new trend definitions in an automated way. We study one such malicious behavior: A malicious user in an online community can launch a Sybil attack [14] by creating a large number of virtual identities. These identities can then work together to provide the owner with some unfair advantage, by outvoting legitimate users in the network. Such Sybil users tend to be connected to many other Sybil users while having a small number of connections to the rest of the network. Given that trends in social networks are highly influential, for instance many companies or individuals invest heavily to get their hashtags into the trending topics in Twitter [33], it is important that the trending topics reported are not biased by the spam of a small number of Sybil nodes. At the other extreme one might be interested in automatically detecting topics that are hot among a clustered community so that suspicious activity can be detected. We claim that using structural trends, one of the gains is to identify or filter topics bolstered by malicious users in a Sybil setting. We note however that, this is not meant as a generic solution for every spam behavior in social networks. Spam in social networks is an important problem that attracted attention of many researchers [6, 20, 41, 39, 15]. Such studies show that malicious behavior in social networks today is not limited to Sybil attacks. The effectiveness of structural trends in detecting or filtering spam under these different models is an open problem we plan to investigate in the future. Though, we would still like to stress that spam detection is a nice side effect rather than a main goal of structural trend analysis.

In order to validate these propositions, we used the 500-node synthetic graph and identified a set of nodes as *Sybil* by randomly selecting a seed and performing a breadth-first search until a number of attack edges are reached. This method of testing Sybil behavior is based on the technique in [42]. Let the set of Sybil nodes be denoted S_{sybil}. Assume that this set of Sybil nodes are interested in a topic T_y and have little interest in other topics whereas the interests of the other users are uniform among all the topics.

We evaluated the relative importance of topic T_y, i.e. point of interest of Sybil nodes, as a traditional, correlated or uncorrelated trend with varying sizes of $|S_{sybil}|$. W.l.o.g. we set topic T_1 as the topic of interest of the Sybil nodes. We answer two questions: 1) For a fixed size of Sybil nodes, how do the p and q values of Sybil nodes for T_1 effect the relative trendiness of T_1 as a traditional, correlated and uncorrelated trend? 2) How does the size of Sybil nodes affect the same metric? In order to answer the former question, a



(a) Effect of Sybil p, q values (b) Effect of # of Sybil nodes

Figure 2: General Influence Spread

set of experiments with increasing p and q values of Sybil nodes for T_1 were performed where the Sybil attack size was set to 10 nodes. The results are presented in Figure 2(a) where the X-axis denotes the setting of the p and q values of Sybil nodes for topic T_1 and Y-axis denotes the importance of T_1 as a *traditional, correlated* and *uncorrelated* trend. "Importance" refers to the number of topics T_1 outranks (including T_1 itself). So when T_1 is the highest ranking topic, its importance is 50 as there are 50 possible topics in the data set. As it can be seen from Figure 2(a) with changing p and q values, correlated score of T_1 is consistently higher than traditional score and traditional score is higher than that of the uncorrelated score. It is also worthwhile to point out that, even with small values of p and q, we can see a breakpoint upon which T_1 becomes significantly more trendy under correlated trendiness, whereas this breakpoint is much later for the other two definitions.

A similar effect is observed in Figure 2(b) where the effect of the number of Sybil nodes in the trendiness of T_1 is given. The X-axis refers to the number of Sybil nodes while the Y-axis demonstrates the same notion as the Y-axis in Figure 2(a). This set of experiments, for a fixed setting of p and q values (For Sybil nodes:$p_{i,1} = q_{i,j,1} = 0.9$ and $p_{i,k} = q_{i,j,k} = 0.01$ for $n_i \in S_{sybil}$, $T_k \in T - T_1$ and $n_j \in N$. As for non-sybil nodes: $p_{i,k} = q_{i,j,k} = 0.0.03$ for $n_i \notin S_{sybil}$, $T_k \in T$ and $n_i \in N$), tests the importance of the number of Sybil nodes and shows correlated trendiness of T_1 is consistently higher than its traditional and uncorrelated trendiness. Also, the jump in *correlated* importance of T_1, which is useful for detecting the suspicious activity, can be observed with a small set of Sybil nodes whereas this jump is seen much later with the other definitions.

4.2 Analysis-Based Value Proposition

Although using the methods introduced in Section 4.1, the value of structural trend definitions can be systematically studied, a verification using real data sets is crucial. We use a large real world data set obtained from Twitter. Leskovec et al. [31] published a data set of 467 million Twitter posts from 20 million users spanning a 7 month period. Author, time and content is available for each tweet. Using the Twitter social network graph published by Kwak et al.[23], we obtained the connections between the users sharing these tweets. We used hashtags to identify topics of tweets. We observed that there were many hashtags that were very similar except for some punctuation details or case differences. We categorize such hashtags under one topic. Although there are over 10 million different hashtags in the Twitter data set, using this technique, we were able to reduce this number to approximately 3.4 million. The social network graph consists of approximately 41.7 million users among those 2.7 million have at least one tweet that includes at least one hashtag. Such nodes have 230 million edges between them whereas the number of edges in the original Twitter social network are 1.47 billion. We now summarize the key findings based on the analysis of this data set.

Question 1: Are traditional trends a good representative of structural trends? Similar to the model-based verification, the *Spearman's correlation coefficient* and *average precision* values were computed to observe how similar structural trends are with traditional trends. As it can be observed from Table 3, spearman correlation of traditional trends and correlated trends is very low indicating that for the entire set of topics their traditional trendiness counts are not a good representative of their *correlated* trendiness. However, we are mostly interested in the similarity of trends, i.e. topics that have a large g score in Equation 2. In order to study the similarity of the trends rather than the similarity of the ranking of the entire set of topics, we calculated *average precision* values for the top-10, 1, 0.1, 0.01 *percentile* traditional topics in detecting the top-10, 1, 0.1, 0.01 *percentile* correlated topics. Top-10, top-1, top-0.1 and top-0.01 percentile correspond to 34633, 3463, 346 and 34 topics accordingly. These values are presented in rows $AP(34633)$, $AP(3463)$, $AP(346)$ and $AP(34)$ and demonstrate that *correlated trends* are substantially different from *traditional trends*. Likewise, *uncorrelated* trends show significant differences from *traditional trends*. Interestingly, adhering to the results obtained in Section 4.1, *uncorrelated trends* are more similar to *traditional trends* than that of *correlated trends*.

Question 2: What is the nature of topics detected using structural trends? Unfortunately, it is very hard to identify *why* a certain Twitter user decides to share a piece of information. It is possible that the user is influenced by *any* of its neighbors as well as external influences such as news media. Therefore, we cannot analyze the Twitter data w.r.t. p and q values. However, we study the structural ties between nodes participating in a trend. The results give interesting insights. As the goal is to detect interesting topics using *structural* trends that would be undetected otherwise, we identified a set of topics in Twitter data set that have a sizable number of mentions though still not ranking as high to be detected as a *traditional* trend (ranking 60^{th} to 100^{th}). Of those topics we identified topics that have a high structural significance compared to their *traditional* significance, i.e., top-10 topics sorted by $R_{corr}(x) - R_{trad}(x)$ (or $R_{uncorr}(x) - R_{trad}(x)$) where $R_{trad}(x)$, $R_{corr}(x)$ and $R_{uncorr}(x)$ corresponds to the *traditional, correlated* and *uncorrelated* ranking of a topic T_x respectively. We observe that, the *correlated* trends result from broadcasts of a relatively small number of users (7125 on average) with a very large number of connections between them (220292 on average), whereas *uncorrelated trends* result from broadcasts from a large number of distinct users (21987 on average) with small number of ties (205764 on average). The details of these topics are provided in Tables 7 and 8 in Section C.

Table 4 demonstrates the results using three topics; #design, #nevertrust, #hhrs. The columns of the table correspond to the hashtag, traditional, correlated and uncorrelated ranking of the hashtag, number of distinct users that used the hashtag and the number of edges between such users respectively. Of those three topics that have similar traditional scores, #nevertrust has a very high uncorrelated score, #hhrs has a very high correlated score and #design is less significant both as a correlated and uncorrelated trend. We can observe that the *correlated* trend #hhrs originate from a small number of nodes with a large number of edges between them whereas the *uncorrelated* trend #nevertrust originate from a large number of distinct users with a much smaller ratio of edges between them. We also give a visual presentation that demonstrates the difference between the trends detected using *correlated*, *uncorrelated* and *traditional* trends. Next we present some visualization results that demonstrate a similar characteristic. We visualize nodes participating in a hashtag and edges between such nodes. Size of the node is proportional to log_2 of number of tweets that node has on that particular hashtag. We use Prefuse, an open-source software

Table 3: Twitter Similarity Statistics

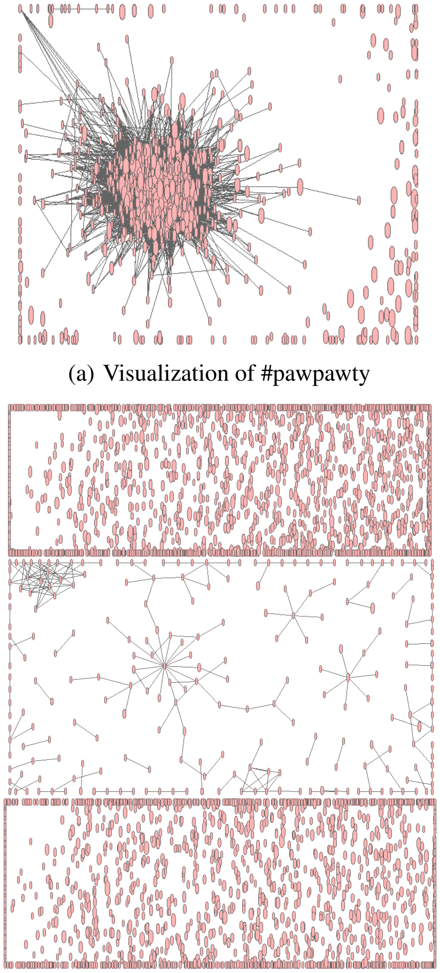
	Correlated	Uncorrelated
ρ	0.14	0.53
$AP(34633)$	0.52	0.9
$AP(3463)$	0.41	0.84
$AP(346)$	0.36	0.61

Table 4: Three Topics From Twitter

hashtag	R_{trad}	R_{corr}	R_{uncorr}	#users	#edges
#design	68	91	164045	15795	280509
#hhrs	75	8	24	5681	381425
#nevertrust	71	779	1	31114	134019

[17], to visualize subgraphs of the Twitter data set, consisting of only nodes that participated in particular hashtags. Unfortunately hashtags mentioned in Table 4 involved too many nodes to be visualized with the current software and memory restrictions. Therefore we provide visualization results for hashtag #pawpawty and #mafiawars which have a total mention between 40000 and 20000, having similar traditional scores. Hashtag #pawpawty ranks 289^{th} while #mafiawars ranks 212^{th}. However, #pawpawty has a high *correlated* importance, ranking 24^{th}, while #mafiawars does not. As we can see from Figure 3, the two trends have completely different behaviors. Unfortunately, due to memory limitation, we were not able to visualize hashtags involving a large number of nodes. Prefuse software started crashing for over 5000 nodes, this is why Figure 3(b) is in fact a combination of 3 subgraphs. The top and bottom parts consist of nodes that have no in or out edges, the middle part consists of nodes that have at least one in or out edge. We can see that #mafiawars has a large number of unconnected nodes, while the opposite is true for #pawpawty. We also note that #pawpawty is a hashtag commonly used to raise money for animal rescue organizations, whereas #mafiawars is commonly used by gamers. This takes us to our next question: Do hashtags with different categorical characteristics have consistently lower or higher structural importance.

In order to answer this question, we analyze 500 hashtags that are categorized into 7 different topics; political, technology, celebrity, games, idioms, movies, music and none. These hashtags and their categories were obtained from a recent study by [**?**]. The authors categorized the top 500 hashtags (defined in terms of number of distinct people tweeting about them) for their data set which overlaps with the data set we use in our study. Therefore these hashtags, though not necessarily top 500 for our data, exist and have significant importance. Our analysis provides some interesting insight as to how people use Twitter to share information. Figure 4(a) demonstrates the CDF of ranking of topics of political category under correlated, uncorrelated and traditional trends. We see that using correlated trends definition, the importance of political hashtags are bolstered. Figure 4(a) also indicates that political hashtags have a high correlated importance indicating that people tend to re-share information shared by their friends, (or simply that homophily leads users to be friends with people with similar political views). However for other categories, such as idioms as demonstrated in Figure 4(b), this is not the case. This phenomenon could have been facilitated by the unique nature of Twitter, which broadcasts tweets of every user. If a user is interested in a specific topic, one can easily obtain the list of tweets under that hashtag. So usage of Twitter might be more centric to the use of this feature rather than social friend following. A different behavior possibly can be observed in other social networks.



(a) Visualization of #pawpawty

(b) Visualization of #mafiawars

Figure 3: Visualization of two hashtags of similar traditional importance in Twitter

5. STRUCTURAL TREND DETECTION

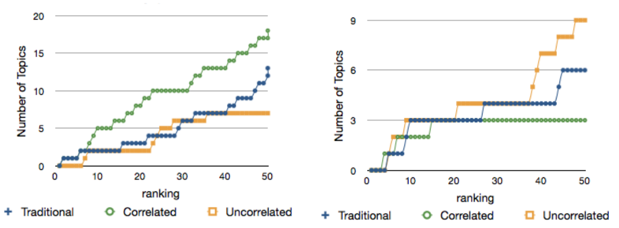
In this section, we provide efficient methods for both *correlated* and *uncorrelated* trend detection. In Section 5.1, we will first give details about the solution for *correlated* trends. Later in Section 5.2, we will provide the details for *uncorrelated* trend detection.

5.1 Correlated Trend Detection

In the following sections, we will first describe the naive solution to the *correlated trend detection*, i.e. computing Equation 2 for each topic. Since this solution is expensive for large social networks with high traffic of information sharing, we next explore ways to gain efficiency. To this end, we propose a sampling based solution in Section 5.1.2. We show that a simple sampling method can be used while still guaranteeing high accuracy, especially for *popular* topics. In order to demonstrate the use of this sampling technique, we reduce the problem of evaluating the importance of each topic with respect to the correlated trendiness notion to a problem of counting local triangles, i.e., counting the number of triangles incident at a given node in a graph G.

5.1.1 Incremental Counting Algorithm

As our main goal is to detect trends, it is crucial to provide incre-



(a) CDF of political hashtag rankings
(b) CDF of idiom hashtag rankings

Figure 4: CDF of ranking of topics of different topics

mental solutions. Therefore algorithms based on the entire data set such as semi-streaming methods [35, 5] are not applicable. In these approaches the data has to be traversed a non-constant number of times. Updates such as receipt of a small number of broadcasts, would necessitate the repetition of the whole process to find new "trends". Instead, we propose using an incremental approach. The approach introduced in this section finds *exact values* and therefore can be computationally expensive, but using the sampling method described in Section 5.1.2, the complexity can be reduced.

Consider the actions need to be taken upon receiving a new tuple $\langle n_l, T_x \rangle$. Assume that until this point the *exact* value of $C_{i,x}$ for each $T_x \in T$ and $n_i \in N$ and the *exact* value for Equation 2 for each T_x are known. Upon the receipt of $\langle n_l, T_x \rangle$, $C_{l,x}$ has to be incremented by 1. The score of T_x should also be updated as:

$$g'(T_x) = g(T_x) + \sum_{n_i \in N_l'} C_{i,x} + \sum_{n_i \in N_l} C_{i,x} \qquad (6)$$

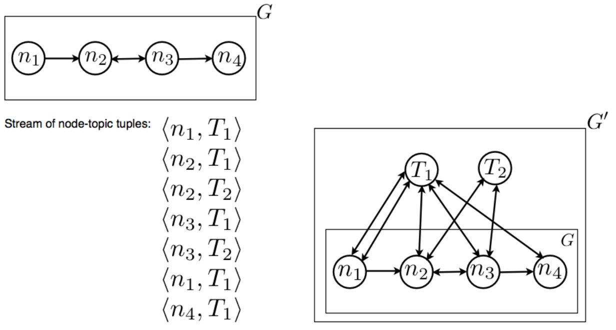
where $g(T_x)$ is the *correlated* score of T_x before receipt of $\langle n_l, T_x \rangle$, $g'(T_x)$ is its score afterwards, $N_i = \{n_j | e_{i,j} \in E\}$ and $N_i' = \{n_j | e_{j,i} \in E\}$. The proof of the correctness of this equation can be found in the Appendix. As is evident from this computation, after receiving tuple $\langle n_l, T_x \rangle$, the "trendiness score" of T_x has to be increased by the sum of all $C_{j,x}$ such that n_j is a neighbor of n_l and $C_{m,x}$ such that n_l is a neighbor of n_m. This requires in the worst case O(n) reads. However, in social networks, only a small fraction of nodes are connected to a large number of nodes. Therefore, in most cases this operation requires a small number of reads. The solution requires using two adjacency lists per node n_i, one to keep track of edges $e_{i,j}$ and another to keep track of edges of the form $e_{j,i}$. Fast access to $C_{i,x}$ for each i and x is needed as well. Therefore a hashtable per topic is used to keep track of the counts of broadcasts per node (n_i being the key to the hashtable H_x for count $C_{i,x}$).

As our ultimate goal is to give an ordered list of *top-k* correlated topics *at each point in time*, in addition to accurately reporting correlated scores per topic we need to provide a sorted representation of the list of *top-k* topics sorted w.r.t. their scores. Therefore, a simple solution uses a sorted structure to keep track of *top-k* topics. In this case the receipt of a new tuple $\langle n_l, T_x \rangle$ might require an update to this structure as well. The naive implementation provides a good solution for small networks with a small number of broadcasts per seconds since it is an O(n + k) solution that is practically even faster because of the power-law properties of social networks. However, the sheer volume of information shared on online social networks today still poses a scalability challenge. A recent report from Twitter announced 3283 tweets per second [34]. Data flow at this scale calls for solutions that sacrifice accuracy for efficiency. We propose a solution based on sampling that provides computational gain while still providing a good level of accuracy.

5.1.2 Counting Local Triangles and Sampling

In this section, we propose our sampling based solution to the scalability challenge of correlated trend detection. As it is much easier to communicate the correctness of the solution in a graph-oriented manner, we will show that the problem of finding *correlated trends* is equivalent to counting local triangles in a multi-graph. Later, we will prove that using sampling this specific problem can be made more efficient.

Consider a social network graph $G = (N, E)$, a set of all topics T and stream of tuples S, where each tuple is in the form: $\langle n_i, T_x \rangle$ s.t. $n_i \in N$ and $T_x \in T$. Let us create a directed multi-graph $G' = (N', E')$ s.t. $N' = N \cup T$ and $E' = \{(u,v) | (u,v) \in E \wedge \langle u,v \rangle \in S \wedge \langle v,u \rangle \in S\}$. The nodes of the network can be categorized into two categories: *topic nodes* $T_x \in T$ and *user nodes* $n_i \in N$. Let us call the edges of the form (n_i, T_x) (or (T_x, n_i)) as *topic edges* and denote this set as E_t. Similarly, edges of the form (n_i, n_j) are *friendship edges* and are denoted by E_f. Clearly $E' = E_t \cup E_f$. In a multi-graph two vertices may be connected by more than one edge. By construction of G', there can be at most one *friendship* edge from a node n_i to n_j. However, there can be an arbitrary number of *topic* edges from a node n_i to T_x (or from T_x to n_i). Any three nodes u, v and w s.t. $(u,v) \in E' \wedge (v,w) \in E' \wedge (w,u) \in E'$ form a *triangle* in G'. $g(T_x)$ score of a topic T_x given in Equation 2 is simply the number of triangles incident to node T_x in G'. Figure 5 gives an example of one such reduction. Note that nodes T_1, n_2, n_3 induce two triangles whereas T_1, n_3, n_4 induce only one triangle due to the fact that (n_2, n_3) is a bidirectional edge whereas (n_3, n_4) is unidirectional. Also T_1, n_1, n_2 induce two triangles precisely because there are two *topic* edges between T_1 and n_1.


G
n_1
n_2
n_3
n_4
Stream of node-topic tuples:
$\langle n_1, T_1 \rangle$
$\langle n_2, T_1 \rangle$
$\langle n_2, T_2 \rangle$
$\langle n_3, T_1 \rangle$
$\langle n_3, T_2 \rangle$
$\langle n_1, T_1 \rangle$
$\langle n_4, T_1 \rangle$
G'
T_1
T_2
G
n_1
n_2
n_3
n_4

(a) A graph G and stream of topic-node tuples S
(b) The multigraph created using G and S

Figure 5: General Influence Spread

After this reduction, the stream of $\langle n_i, T_x \rangle$ tuples can be observed as the incoming *topic* edges of G'. Next, we will demonstrate that given the entire graph G is available and only E_t, the *topic edges* are sampled, correlated trendiness for topics can be accurately predicted. The procedure is straightforward and the sampling method resembles of the one introduced in [36]: Create a directed multi-graph $G'' = (N'', E'')$ s.t. $N'' = N'$ and $E'' = \{(u,v) | (u,v) \in E\}$. For each incoming tuple $\langle n_i, T_x \rangle$, which corresponds to *topic* edges (n_i, T_x) and (T_x, n_i) in G', flip a coin with bias p_s. With p_s probability, we keep *both* (n_i, T_x) and (T_x, n_i) edges by setting $E'' = E'' \cup (n_i, T_x) \cup (T_x, n_i)$ and discard them both otherwise. Number of triangles involving T_x in G' can be estimated as $X_x = Count_x / {p_s}^2$, where $Count_x$ denotes the number of triangles involving T_x in G''. We can guarantee that the number of triangles calculated based on the sampled data is a good approximation of the actual number of triangles. Specifically, the probability that the prediction X_x is off by $\varepsilon\Delta_x$ is upper-bounded by the following equation:

$$Pr(|X_x - \Delta_x| \geq \varepsilon\Delta_x) \leq \frac{Var(X_x)}{\varepsilon^2{\Delta_x}^2} \leq \frac{({p_s}^2 - {p_s}^4)}{{p_s}^4\varepsilon^2\Delta_x} + 2\alpha_x \frac{({p_s}^3 - {p_s}^4)}{{p_s}^4\varepsilon^2{\Delta_x}^2} \tag{7}$$

where Δ_x is the actual number of triangles involving T_x, α_x is the number of pairs of triangles that involve T_x and are not edge disjoint and p_s is the rate of sampling. The proof of correctness of Equation 7 is provided in Section A. As is evident from Equation 7 the quality of the estimate depends on the number of triangles as well as the number of edge-disjoint triangles. Since the number of multi-edges has a big effect on this property, the quality of the estimate depends on number of times a specific user mentions a specific topic. As this number gets increasingly large, the quality of the estimate degrades. However the estimate gets quadratically better with increasing Δ_x and getting only linearly worse with the α_x which is smaller so the estimate is still better for "trendy" topics.

5.2 Uncorrelated Trend Detection

Similar to *correlated* trends, *uncorrelated* trends can be reduced to counting local triangles in a multi-graph. Consider a social network graph $G = (N, E)$, a set of all topics T and stream of tuples S, where each tuple is in the form: $\langle n_i, T_x\rangle$ s.t. $n_i \in N$ and $T_x \in T$. Let us create a multi-graph $G' = (N', E')$ s.t. $N' = N \cup T$ and $E' = \{(u,v) | (u,v) \notin E \wedge (u,v) \in S\}$. $h(T_x)$ score of a topic T_x, is simply the number of triangles incident to node T_x in G'. In this setting, the tuples $\langle n_i, T_x\rangle$ that are being streamed can be seen as the edges of the multi-graph G'. As demonstrated in Section 5.1.2, this problem can be efficiently approximated by sampling.

Since an online algorithm is a requirement, the *uncorrelated* trendiness score of topics should be incrementally updated and reported. The exact increase can be calculated in the following way:

$$h'(T_x) = h(T_x) + \sum_{n_i \in (N - n_l - N_l)} C_{i,x} + \sum_{n_i \in (N - n_l - N_l')} C_{i,x} \tag{8}$$

where $h(T_x)$ is the *uncorrelated* score of T_x before receipt of the new tuple $\langle n_l, T_x\rangle$, $h'(T_x)$ is its score after the receipt of the tuple, $N_i = \{n_j | e_{i,j} \in E\}$and $N_i' = \{n_j | e_{j,i} \in E\}$. Upon receiving tuple $\langle n_l, T_x\rangle$, the *uncorrelated trendiness* score of T_x has to be increased by the sum of all $C_{j,x}$ such that n_j is not a neighbor of n_l and $C_{m,x}$ such that n_l is not a neighbor of n_m. This requires in the worst case $O(n)$ reads. Unfortunately unlike the computation necessary *correlated trendiness*, this operation in most cases requires close to n reads. However, a simple realization results in an efficient solution that would make use of the solution provided for *correlated trends*. By keeping track of *traditional trendiness* score, $f(T_x)$, for each topic T_x, the update on $h(T_x)$ can be computed as: $2 * f(T_x) - \sum_{n_j \in N_i} C_{j,x} - \sum_{n_j \in N_i'} C_{j,x}$. This way, one can still make use of the power-law degree distribution of social networks which means a small number of reads per update operation.

5.3 Sampling on Twitter

Our goal is to provide a ranked list of top-k topics for both trend definitions. Therefore, we performed experiments on the Twitter data set introduced in Section 4.2 to compute *average precision*(*AP*) of sampled data for both *correlated* and *uncorrelated* top-k lists for different values of sampling parameter p_s (0.5,0,2,0.1, 0.01 and 0.005) and k. Figure 6, which provides the results for *correlated* trends, shows that top-34 *correlated* trend detection is largely robust to the sampling parameter, i.e. even for a small value of $p = 0.005$ where approximately 1 out of 200 tuples is processed, *AP* lies above 0.93. This is not the case for top-34633 topics where *AP* degrades largely with decreasing p_s. This is mostly due to the large number of *tail* topics that are unpopular and have close-to-zero values. This behavior therefore is to be expected considering that sampling has low accuracy for unpopular topics as is shown in Equation 7. Note, however, unpopular topics are of little interest for trend detection. Results for *uncorrelated* trends are similar to that of *correlated* trends and is provided in Appendix. Interestingly, *uncorrelated* trend detection is more robust to sampling. That is to be expected as the quality of sampling is higher for larger values of exact number of triangles as given in Equation 7 and due to the sparsity of social network graphs number of triangles induced from *uncorrelated* trendiness tend to be larger than that of *correlated* trendiness. As could be expected, a inear speed-up is observed w.r.t $1/p_s$. We refer the reader to Section C for the figures.

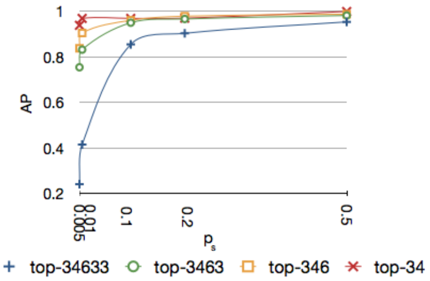


Figure 6: Average Precision of sampling for correlated trends

6. CONCLUSION

In this paper, we introduced new methods for analysis of trends in social network that incorporate the structural properties of the network. We propose two new structural trend definitions called *correlated* and *uncorrelated* trends that leverage from the friendship information to detect interesting topics that would be undetected using traditional trend definitions. We introduced a novel information diffusion model called *Independent Trend Formation Model (ITFM)* that captures the diffusion of an arbitrary number of topics in a social network. Using *ITFM*, we identified properties of *structural* trends that distinguish them from *traditional* trends. We also show that this difference in nature corresponds to interesting activity, including detection (or filtering) of Sybil activity. We also performed experiments on a large scale real social network data from Twitter with 41.7 million nodes and 417 million posts. Results obtained from these experiments adhere to the results obtained using the *ITFM* model which in addition to supporting the value proposition of *structural* trends, also indicates the *ITFM* model reflects real social network behavior.

Detection of structural trends is inherently harder than the traditional trend detection. Therefore we proposed a sampling technique that provides computational gain while still being within an acceptable error bound. Experiments performed on the large-scale Twitter data set show that even with a small sampling rate of 0.005, the average precision lies above 0.93 for *correlated* trends while keeping a perfect average precision of 1 for *uncorrelated* trends. As future work, we will study more general structural trend definitions that explore the space between the two extremes introduced in this work, such as a group of c connected people discussing a topic (or a group of people where every node is connected to at least c other nodes). We also plan to investigate other methods of approximation for the structural trend detection.

7. REFERENCES

[1] J. Allan, editor. *Topic detection and tracking: event-based information organization*. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[2] N. Alon and J. Spencer. *The probabilistic method.* Wiley-Interscience, 2000.

[3] N. Alon, R. Yuster, and U. Zwick. Finding and counting given length cycles. *Algorithmica*, 17:209–223, 1997.

[4] Z. Bar-Yossef, R. Kumar, and D. Sivakumar. Reductions in streaming algorithms, with an application to counting triangles in graphs. In *SODA '02*, pages 623–632, 2002.

[5] L. Becchetti, P. Boldi, C. Castillo, and A. Gionis. Efficient semi-streaming algorithms for local triangle counting in massive graphs. In *KDD '08*, pages 16–24, 2008.

[6] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. Detecting spammers on twitter. In *Proceedings of the 7th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, 2010.

[7] S. Bharathi, D. Kempe, and M. Salek. Competitive influence maximization in social networks. In *WINE*, pages 306–311, 2007.

[8] C. Budak, D. Agrawal, and A. El Abbadi. Limiting the spread of misinformation in social networks. Technical Report UCSB/CS-2008-02, UCSB, 2010.

[9] L. S. Buriol, G. Frahling, S. Leonardi, A. Marchetti-Spaccamela, and C. Sohler. Counting triangles in data streams. In *PODS '06*, pages 253–262, 2006.

[10] T. Carnes, C. Nagarajan, S. M. Wild, and A. van Zuylen. Maximizing influence in a competitive social network: a follower's perspective. In *ICEC '07*, pages 351–360. ACM, 2007.

[11] M. Charikar, K. Chen, and M. Farach-Colton. Finding frequent elements in data streams. In *ICALP'02*, pages 693–703, 2002.

[12] G. Cormode and S. Muthukrishnan. What's Hot and What's Not: Tracking Most Frequent Items Dynamically. *TODS'05*, 30(1):249–278, 2005.

[13] E. D. Demaine, A. López-Ortiz, and J. I. Munro. Frequency estimation of internet packet streams with limited space. In *ESA'02*, volume 2461, pages 348–360, 2002.

[14] J. R. Douceur. The sybil attack. In *IPTPS*, pages 251–260, 2002.

[15] C. Grier, K. Thomas, V. Paxson, and M. Zhang. @spam: the underground on 140 characters or less. In *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, pages 27–37, New York, NY, USA, 2010. ACM.

[16] S. Havre, B. Hetzler, and L. Nowell. ThemeRiver: visualizing theme changes over time. In *InfoVis 2000*, pages 115–123, 2000.

[17] J. Heer, S. Card, and J. Landay. Prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430. ACM, 2005.

[18] Tweetstats. `http://tweetstats.com/trends`.

[19] Trendistic. `http://trendistic.com/`.

[20] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer. Social phishing. *Commun. ACM*, 50:94–100, October 2007.

[21] H. Jowhari and M. Ghodsi. New streaming algorithms for counting triangles in graphs. In *COCOON'05*, pages 710–716, 2005.

[22] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD'03*, pages 137–146, 2003.

[23] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? In *WWW '10*, pages 591–600, 2010.

[24] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *KDD '09*, pages 497–506, 2009.

[25] G. S. Manku and R. Motwani. Approximate frequency counts over data streams. In *VLDB'02*, pages 346–357, 2002.

[26] C. D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[27] J. Maritz. *Distribution-free statistical methods*. Chapman & Hall/CRC, 1995.

[28] A. Metwally, D. Agrawal, and A. El Abbadi. An integrated efficient solution for computing frequent and top-k elements in data streams. *TODS'06*, 31(3):1095–1133, 2006.

[29] A. Sala, L. Cao, C. Wilson, R. Zablit, H. Zheng, and B. Y. Zhao. Measurement-calibrated graph models for social network experiments. In *WWW '10*, pages 861–870, 2010.

[30] J. Sankaranarayanan, H. Samet, B. E. Teitler, M. D. Lieberman, and J. Sperling. Twitterstand: news in tweets. In *GIS '09*, pages 42–51, 2009.

[31] Snap: Network datasets: 476 million twitter tweets. `http://snap.stanford.edu/data/twitter7.html`.

[32] B. E. Teitler, M. D. Lieberman, D. Panozzo, J. Sankaranarayanan, H. Samet, and J. Sperling. Newsstand: a new view on news. In *GIS '08*, pages 1–10, 2008.

[33] How much is a tweet worth? $500, says toyota. `http://mashable.com/2010/12/13/toyota-shareathon/`.

[34] Another big record: Part deux. `http://blog.twitter.com/2010/06/another-big-record-part-deux.html`.

[35] C. E. Tsourakakis. Fast counting of triangles in large real networks without counting: Algorithms and laws. In *ICDM '08*, pages 608–617, 2008.

[36] C. E. Tsourakakis, U. Kang, G. L. Miller, and C. Faloutsos. Doulion: counting triangles in massive graphs with a coin. In *KDD '09*, pages 837–846, 2009.

[37] Twitter. `http://twitter.com/`.

[38] Why twitter hashtags and trending topics are useless to marketers. `http://blog.hubspot.com/blog/tabid/6307/bid/4694/Why-Twitter-Hashtags-and-Trending-Topics-Are-Useless-to-Marketers.aspx`.

[39] A. H. Wang. Don't follow me - spam detection in twitter. In *SECRYPT*, pages 142–151, 2010.

[40] Twitter: We are not keeping wikileaks out of trending topics. `http://mashable.com/2010/12/06/wikileaks-twitter-censorship/`.

[41] S. Yardi, D. M. Romero, G. Schoenebeck, and D. Boyd. Detecting spam in a twitter network. *First Monday*, 15(1), 2010.

[42] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. Sybilguard: defending against sybil attacks via social networks. *SIGCOMM Comput. Commun. Rev.*, 36:267–278, 2006.

Table 5: Definitions of symbols and Acronyms

Symbol	Definition
$G=(N,E)$	social network graph
S	stream of node-topic tuples
T	Topic nodes in G' (induced from the set of all possible topics)
$G'=(N',E')$	multi-graph induced using G, T and S
N'	$=N\cup T$
E'	$=E_f\cup E_t$ where E_f is the set of friendship edges and E_t is the set of topic edges
G''	multi-graph after sampling
Δ_x	number of triangles involving T_x in $G\prime$
$\delta_{x,j}$	indicator variable, $\delta_{x,j}=1$ if j^{th} triangle of T_x exists in G'' and $\delta_{x,j}=0$ otherwise ($j=1,...,\Delta_x$)
X_x	estimate of number of triangles after sampling (computed as $Count_x/p^2$, where $Count_x$ denotes the number of triangles involving T_x in G'')
p_s	sampling probability, i.e. probability that a topic edge in G' exists in G''

APPENDIX

A. PROOF OF QUALITY OF SAMPLING

In this section we provide the proof of Equation 7 which provides guarantees for the error bound of sampling in identifying number of local triangles in G' where a certain subset of the edges (E_t) are sampled. We refer the reader to Section 5.1.2 for construction of G' and G'' and provide an overview of notations introduced in Section 5.1.2 as a guideline since these notations will be used used throughout the proof.

Here we will prove that given the entire set of N' and E_f available and only E_t, i.e. *topic edges* are sampled, number of local triangles involving a specific *topic node* T_x can be accurately estimated. We start by studying the mean and variance of the number of triangles detected on the sampled data and derive bounds on the expected number of triangles detected with respect to the actual number of triangles. Table 5 lists the symbols and acronyms used throughout this section. We first show that the expected number of triangles involving T_x in G'' is Δ_x, i.e. the number of triangles with an end node T_x in G'.

THEOREM A.1. *The expected value of X_x in G'' is equal to the actual number of triangles in G' (or equivalently the score of topic T_x), i.e. $E[X_x]=\Delta_x$.*

PROOF. Random Variable X_x is the sum of indicator variables for topic T_x multiplied by $(1/p_s)^2$. Therefore, $E[X_x]=E[\sum_{j=0}^{\Delta_x}\delta_{x,j}/p_s{}^2]=$ □ $\sum_{j=0}^{\Delta_x}E[\delta_{x,j}/p_s{}^2]=1/p_s{}^2\sum_{j=0}^{\Delta_x}E[\delta_{x,j}]=1/p_s{}^2\sum_{j=0}^{\Delta_x}p_s{}^2=\Delta_x$ □ □

Using Chebyshev's inequality that states $Pr(|X_x-\Delta_x|\geq\varepsilon\Delta_x)\leq\frac{Var(X_x)}{\varepsilon^2\Delta_x{}^2}$, one can provide guarantees on the accuracy of X_x in predicting the actual Δ_x values. In order to do so, we now study the variance of variable X_x.

THEOREM A.2. *The variance of X_x, the random variable denoting the estimate of triangles involving T_x based on the sampled data, is equal to:*

$$Var(X_x)=\frac{\Delta_x(p_s{}^2-p_s{}^4)+2\alpha_x(p_s{}^3-p_s{}^4)}{p_s{}^4}$$

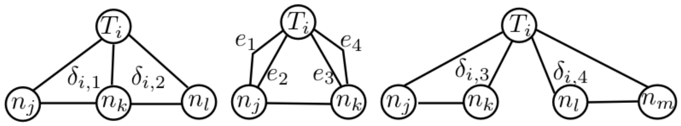


Figure 7: Cases to be considered for variance

where α_x is the number of pairs of triangles that involve T_x and are not edge disjoint.

PROOF. X_x is a sum of indicators that a certain triangle involving T_x survives after sampling. These indicators are not independently distributed. Consider two triangles denoted by indicator variables $\delta_{x,j}$ and $\delta_{x,l}$ where $j\neq l$ (j^{th} and l^{th} triangle involving T_x). Such two triangles cannot share all three edges as they are distinct triangles. They can neither share two *friendship edges* since two user nodes can have up-to 1 edge between them. They cannot share two *topic edges* either since in this case the triangles again would be identical as two triangles sharing two topic edges would also have to share the friendship edge as there is up-to 1 edge between two user nodes. Eliminating such possibilities, there are four possible cases to be considered: 1) They share one *topic edge* ($\delta_{i,1}$ and $\delta_{i,2}$ in Figure 7), 2) They share one *friendship edge* 3) They share one *friendship* and one *topic edge* or 4) They share no edges ($\delta_{i,3}$ and $\delta_{i,4}$ in Figure 7). Figure 7 lists these possible scenarios of how two such indicators might (or not) be dependent. For case 2) and 4), the two indicators would be independent as friendship edges are not sampled. For cases 1) and 3), the two indicator variables both are dependent on the *topic edge* “surviving”. Let number of cases of the form 1) or 3) be α_x for topic T_x. The variance of X can be computed as:

$$Var(X_x)=Var(\frac{1}{p^2}\sum_{j=1}^{\Delta_x}\delta_{x,j})=\frac{1}{p^4}\sum_{j=1}^{\Delta_x}\sum_{l=1}^{\Delta_x}Cov(\delta_{x,j},\delta_{x,l})$$

There are $\Delta_x{}^2$ terms in this summation. Δ_x of these terms are the variances of indicator variables. Since there are α_x of cases where two indicator variables are dependent on each other (share one *topic edge*), the covariance for α_x out of $\binom{\Delta_x}{2}$ pairs of indicator variables is: $Cov(\delta_{x,j},\delta_{x,l})=p_s{}^3-p_s{}^4$. $Cov(\delta_{x,m},\delta_{x,o})=p_s{}^4-p_s{}^4=0$ for the rest $\binom{\Delta_x}{2}-\alpha_x$ terms. Therefore the variance can be computed as:

$$Var(X_x)=\frac{1}{p_s{}^4}(\Delta_x(p_s{}^2-p_s{}^4)+2\alpha_x(p_s{}^3-p_s{}^4))$$

□

Therefore, using Chebyshev's inequality [2] and substituting results from Theorem A.2, we can provide error bounds on the number of triangles detected on the sampled data in the following way:

$$Pr(|X_x-\Delta_x|\geq\varepsilon\Delta_x)\leq\frac{Var(X_x)}{\varepsilon^2\Delta_x{}^2}\leq\frac{(p_s{}^2-p_s{}^4)}{p_s{}^4\varepsilon^2\Delta_x}+2\alpha_x\frac{(p_s{}^3-p_s{}^4)}{p_s{}^4\varepsilon^2\Delta_x{}^2}$$

This proves the correctness of Equation 7.

B. INCREMENTAL CORRELATED AND UNCORRELATED SCORE UPDATES

In this section we prove the correctness of Equations 6 and 8 which identify how *correlated* and *uncorrelated* scores of a topic T_x need to be update upon receipt of a tuple $\langle n_l, T_x \rangle$.

B.1 Correlated Score Update

After receiving tuple $\langle n_l, T_x \rangle$, the "trendiness score" of T_x has to be increased by the sum of all $C_{j,x}$ such that n_j is a neighbor of n_l and $C_{m,x}$ such that n_l is a neighbor of n_m as given Equation 6. Now, we will prove the correctness of this statement. Let the counts per node be denoted with C before receipt of the new tuple and C' after receipt of the new tuple. Keeping in mind that the only C' value changed is $(C_{l,x})'$, the exact increase can be calculated in the following way:

$$\begin{aligned}
g'(T_x) &= \sum_{\substack{n_i \in N \\ n_j \in N_i}} C_{i,x}' \cdot C_{j,x}' \\
&= \sum_{\substack{n_i \in N - n_l \\ n_j \in N_i - n_l}} C_{i,x}' \cdot C_{j,x}' + \sum_{\substack{n_i = n_l \\ n_j \in N_i}} C_{i,x}' \cdot C_{j,x}' + \sum_{\substack{n_j = n_l \\ n_i \in N_j'}} C_{i,x}' \cdot C_{j,x}' \\
&= \sum_{\substack{n_i \in N - n_l \\ n_j \in N_i - n_l}} C_{i,x} \cdot C_{j,x} + \sum_{n_j \in N_l} (C_{l,x} + 1) \cdot C_{j,x} + \sum_{n_i \in N_l'} C_{i,x} \cdot (C_{l,x} + 1) \\
&= \sum_{\substack{n_i \in N \\ n_j \in N_i}} C_{i,x} \cdot C_{j,x} + \sum_{n_i \in N_l'} C_{i,x} + \sum_{n_i \in N_l} C_{i,x} \\
&= g(T_x) + \sum_{n_i \in N_l'} C_{i,x} + \sum_{n_i \in N_l} C_{i,x}
\end{aligned}$$

where $g(T_x)$ is the *correlated* score of T_x before receipt of $\langle n_l, T_x \rangle$, $g'(T_x)$ is its score afterwards, $N_i = \{n_j | e_{i,j} \in E\}$ and $N_i' = \{n_j | e_{j,i} \in E\}$.

B.2 Uncorrelated Score Update

According to Equation 8, upon receiving tuple $\langle n_l, T_x \rangle$, the *uncorrelated trendiness* score of T_x has to be increased by the sum of all $C_{j,x}$ such that n_j is not a neighbor of n_l and $C_{m,x}$ such that n_l is not a neighbor of n_m. Now we will prove this statement. Similar to Section B.1, the counts per node are denoted with C before receipt of the new tuple and C' after receipt of the new tuple.

$$\begin{aligned}
h'(T_x) &= \sum_{\substack{n_i \in N \\ n_j \in N_i^c}} C_{i,x}' \cdot C_{j,x}' \\
&= \sum_{\substack{n_i \in (N - n_l) \\ n_j \in (N_i^c - n_l)}} C_{i,x}' \cdot C_{j,x}' + \sum_{\substack{n_i = n_l \\ n_j \in N_i^c}} C_{i,x}' \cdot C_{j,x}' + \sum_{\substack{n_j = n_l \\ n_i \in N_j'^c}} C_{i,x}' \cdot C_{j,x}' \\
&= \sum_{\substack{n_i \in (N - n_l) \\ n_j \in (N_i^c - n_l)}} C_{i,x} \cdot C_{j,x} + \sum_{n_j \in N_i^c} (C_{l,x} + 1) \cdot C_{j,x} + \sum_{n_i \in N_l'^c} C_{i,x} \cdot (C_{l,x} + 1) \\
&= \sum_{\substack{n_i \in N \\ n_j \in N_i^c}} C_{i,x} \cdot C_{j,x} + \sum_{n_i \in (N - n_l - N_l)} C_{i,x} + \sum_{n_i \in (N - n_l - N_l')} C_{i,x} \\
&= h(T_x) + \sum_{n_i \in (N - n_l - N_l)} C_{i,x} + \sum_{n_i \in (N - n_l - N_l')} C_{i,x}
\end{aligned}$$

where $h(T_x)$ is the *uncorrelated* score of T_x before receipt of the new tuple $\langle n_l, T_x \rangle$, $h'(T_x)$ is its score after the receipt of the tuple, $N_i = \{n_j | e_{i,j} \in E\}$, $N_i' = \{n_j | e_{j,i} \in E\}$, $N_i^c = N - n_i - N_i$ and $N_i'^c = N - n_i - N_i'$.

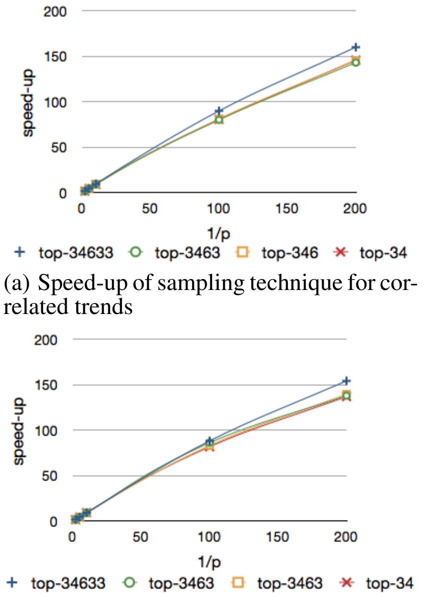
Table 6: Model Similarity Statistics

p	$\rho_{trad-corr}$	$\rho_{trad-uncorr}$	AP_{corr}	AP_{uncorr}
0.1	0.763	0.988	0.144	0.571
0.3	0.518	0.993	0.079	0.672
0.5	0.401	0.996	0.062	0.737

Table 7: Uncorrelated trends in Twitter that are Traditionally Insignificant

hashtag	R_{uncorr}	R_{trad}
#twitter	14	80
#wheniwaslittle	22	88
#5	11	79
#MLB	10	78
#rememberwhen	8	77
#photography	4	73
#hc09	5	74
#health	2	72
#nevertrust	1	71

C. FURTHER RESULTS OF EXPERIMENTS



(a) Speed-up of sampling technique for correlated trends

(b) Speed-up of sampling technique for uncorrelated trends

Figure 8: Speed-up of sampling

Here we provide Table 6 that demonstrates the effect of increasing p values (probability that a node in the network discusses a topic independent from its neighbors). The analysis in summary is provided in Section 4.

We have provided the figures that summarize the results of accuracy experiments for *correlated* trends on the Twitter data set in Section 5.3. As we noted before, the behavior of *uncorrelated* trends is similar to that of *correlated* trends. We also noted that *uncorrelated* trends are more robust to sampling while providing a possible reasoning for this behavior. In this section, we provide the summary of the experiments for *uncorrelated* trends accuracy in Figure 9 for completeness. Similar to Figure 6, the X-axis denotes the rate of sampling whereas the Y-axis denotes the *average precision* for the given sampling ratio. Sampling ratio values used

Table 8: Correlated trends in Twitter that are Traditionally Insignificant

hashtag	R_{corr}	R_{trad}
#politics	58	85
#bb11	45	82
#ocra	43	87
#green	35	81
#freemediave	40	97
#nieuws	27	86
#TCOT	30	89
#digg	15	76
#hhrs	8	75

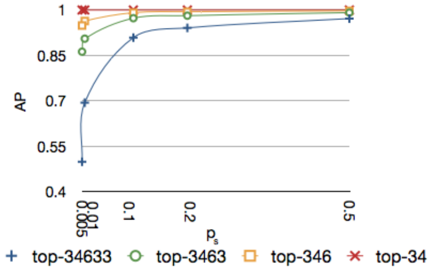


Figure 9: Average Precision of sampling technique for uncorrelated trends

were $p = 0.5, 0, 2, 0., 0.01$ and 0.005. For top-34 topics, for all sampling ratios, even for 0.005, we observe a perfect *average precision* of 1 while this value degrades rapidly for top-34633 *uncorrelated* topics.

As discussed in Section 5.3, sampling provides a linear speed-up. For completeness here we provide the figures that summarize the timing of experiments on the Twitter data set. Figure 8(a) provides the results for *correlated* trends, whereas Figure 8(b) provides the results for *uncorrelated* trends. For both figures, the X-axis denotes the inverse of sampling ratio $(1/p_s)$ and Y-axis provides the speed-up, i.e. the ratio of the time it takes for the exact solution to the time it takes for sampling method to process the entire data set.