

Modeling a Web Forum *Ecosystem* into an Enriched Social Graph

Tarique Anwar¹ and Muhammad Abulaish²

¹ Centre for Computing and Engineering Software Systems
Swinburne University of Technology, Melbourne, VIC 3122, Australia
tanwar@swin.edu.au

² Department of Computer Science
Jamia Millia Islamia (A Central University), New Delhi 25, India
mAbulaish@jmi.ac.in

Abstract. This paper considers the community interactions in online social media (OSM) as an *OSM ecosystem* and addresses the problem of modeling a Web forum ecosystem into a social graph. We propose a text mining method to model cross-thread interactions and interests of users in a Web forum ecosystem to generate an enriched social graph. In addition to modeling *reply-to* relationships between users, the proposed method models message-similarity relationship to keep track of all similar posts resulting out of deviated discussions in different threads. Although, the proposed graph-generation method considers a *reply-to* relation as the primary means of linkage, it establishes links between clusters of similar posts instead of links between individual users, and the linkages between users can be derived from the existing linkages between clusters. The method starts with linking posts in each thread individually by identifying *reply-to* relationships, and applies an agglomerative clustering algorithm based on similarity of posts across the forum to group all posts into different clusters. Finally, relations between each pair of individual posts are mapped to create a link between clusters containing the posts. As a result, the generated social graph resembles a network of clusters that can also be presented at the granule of users who authored the posts to generate a social network of forum users, and at the same time it keeps information for all other users with similar interests.

Keywords: Social media mining; Web forum ecosystem; Social graph generation; Agglomerative clustering.

1 Introduction

Since the inception of Web 2.0, it is increasingly getting crowded with Web users and explosive contents generated by them at a tremendous rate, which characterizes the Web as extremely dynamic and diverse in nature [6]. Nowadays, Web 2.0 applications are endorsing a paradigm shift in the way contents are generated on the Web [29], where users are getting space to generate contents by themselves. A significant percentage of Web users are frequently participating

in various ways to generate Web contents [21]; social networking sites (SNS) such as facebook, twitter, myspace, etc., being the most common of them are intruding rapidly into our lives [24]. Thus, it conduces us to categorize Web contents into the *proprietary contents*, adhering to some well-defined structure, and the *user-generated contents*, which are highly unstructured and noisy [11]. The group of Web-based applications (a division of cyberspace) that build on the ideological and technological foundations of Web 2.0, and allow the creation and exchange of user-generated contents is said to be *online social media* (OSM) [23].

1.1 The OSM *Ecosystem*

Even though the term contains the word “media”, it has very little to do with the traditional information media. Rather, it is more inclined towards the other word “social” (derived from the ties of social relationships) and provides a mechanism for the audience to socialize themselves by mingling with others [9]. OSM is evolving as a powerful tool for people to connect, communicate and interact globally on topics of common interest which take place in various forms ranging from complicated and obscured ones to simple and conspicuous ones. Some instances of these interactions are, i) posting a comment on a facebook update, ii) liking a link shared on a friend’s facebook wall, iii) following someone or being followed by someone on twitter, iv) commenting or replying a blog post, v) participating in a discussion thread on a Web forum, vi) liking or disliking a youtube video, and so on. Even a layman can easily notice the conspicuous relationships in the above instances, but other obscured relationships in them can hardly be noticed even by an expert analyst. For example, let us assume three social media users, u_1 , u_2 and u_3 . Suppose u_1 comments on a thread initiated by u_2 , and u_3 is the user to whom u_2 replies every time she asks any question in a thread. In this scenario, the relationships between u_1 and u_2 as well as u_2 and u_3 are comparatively much more noticeable than the one between u_1 and u_3 . Usually the relationship that an analyst tries to accentuate depends on the type of interaction being focused, where interaction type can be any of the above mentioned or similar instances. These kinds of relationships established among users on the Web create a healthy, social and collaborative *ecosystem* for various community practices. In [21], Jones and Fox observed that nowadays people use the Internet more often to socialize themselves through social media than other activities. For example, when a person without adequate technical knowledge about cars, finds some fault in the gearbox of his car, he initiates a thread on a forum explaining its unexpected symptoms and asks for helpful suggestions. Very soon the thread gets multiple replies by unknown users who share their own experiences with a similar problem and suggest probable solutions that could be helpful to him. In one perspective, the replies in the thread are nothing more than an assistance to the thread initiator, which is clearly visible to all. Can the series of replies, personal experiences and suggestions also be meaningful for any other reason and/or other person? A ponder on this question brings about a considerable number of other perspectives beyond the only apparent

one, that makes us realize the importance of such interactions. The general objectives in this area span over characterizing user behaviors and interactions, analyzing established social relationships among them, and extracting information from the text contents of actual discussions. However, Choudhury *et al.* [10] pointed out that the excitement of information extraction researchers resulting from the overwhelming and explosive growth in user generated Web contents is overshadowing two authoritative and related problems in this area. First one is the *inference problem*, which states that the real social relationships or ties always remain obscured and therefore must be deduced from the unobscured events to bring it into notice. In the example mentioned above, the obscured tie between u_1 and u_3 is to be deduced from their unobscured interactions with u_2 . The other problem pointed out is the *relevance problem* according to which a social network actually is a blend of multiple social networks, each one based on a different definition of relationship and therefore relevant to a different social process.

1.2 Web Forum as an OSM

Despite the fact that Social Networking Sites (SNSs)³ are the most popular online sites [24], there are numerous other ways in which a user participates in OSM activities, e.g., through Web forums, blogs, wikis, bookmarks, diggs, RSS, and so on, where each one has its own distinctive role and effect on the relationship being established. Unlike other OSM⁴, *Web forums* or *discussion boards* provide a platform for formal, vivid and dynamic discussions among an unrestricted number of participants. Figure 1 shows a typical ecosystem formed by user interactions in a Web forum. In this folksonomy, discussions are started by its members in the form of a discussion thread with a title and an entry message post. Viewers of this thread annotate their own opinions or replies to the thread and thus the system keeps on evolving as the number of posts grow in the thread. Starting with an equilibrium state of no posts, it goes through a disequilibrium state once a message is posted as a response, in which the community interacts answering the preceding and following messages. It reaches back to the equilibrium state after all the commentators finish up presenting their views and no further message is appended to the thread [5]. During the course of discussions, the interactions in the form of replies and responses stir to establish some relationship among the unknown users. In the example of fault in the car mentioned in section 1.1, the thread initiator develops a relationship of trust with those replying to him; however the confidence level of trust depends on the *structure* of interactions (replies) and the *relevance* of the message contents to the thread title and entry post. Nevertheless, the unrestricted ordered growth of intertwined posts in a thread with not much support to identify a *reply-to* relationship makes it extremely complicated to trace its actual interaction

³ SNS interactions are usually casual in nature with short and frequent communications

⁴ Each type of OSM has its own distinctive role and effect on the relationship being established.

structure. After a thread is initiated it starts springing up linearly, but with time as it involves more and more participants, the distinct view of each participant in response to a distinct post very often transforms it to complex multi-threaded structure [27]. It has been found that an interaction structure coordinate with social media analysis in a variety of ways [12], like identifying user roles, their social values and the social community structure [22], establishing *ties* among users [13], and so on. The inherent complexities of thread structures and user interactions as well as lack of functionalities in producing organized information in Web forums have actuated research on tracing interaction among users in a forum [12, 25].

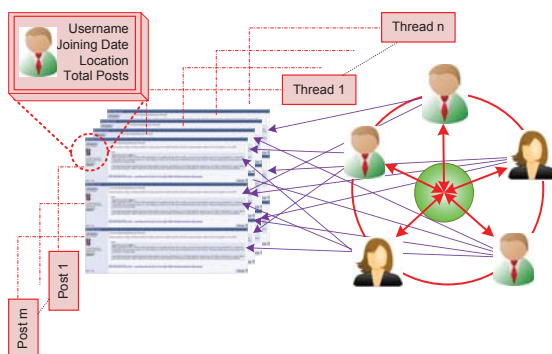


Fig. 1. A Web forum *ecosystem*

1.3 Contributions

A common phenomenon observed in online threaded discussions is that they usually start from a specific topic, but as they grow with more posts, their context goes on deviating from its actual title [15]. Very often a deviated discussion is found to be overlapping with a different thread in the forum. A person replying to the deviated post in one thread is very much likely to reply the similar posts in other threads if he comes to know about this kind of thread overlaps. The state-of-the-art research makes it very clear that the *reply-to* relationships play a prime role in interaction graph generation [12]. But in case of a deviated discussion, a simple *reply-to* relationship fails to capture the relation between a reply-post in a thread, and the posts in other threads which are similar to the post to which the former is replied. This paper basically addresses the problem of modeling a Web forum ecosystem into a social graph. We propose a novel enriched social graph generation method, which (in addition to identifying *reply-to* relationships) identifies message-similarity relationship to keep track of all similar posts resulting out of deviated discussions and thus models cross-thread community interactions and interests. The proposed method still considers *reply-to*

relations as the primary means of linkage, but rather than establishing links between users, links are established between clusters of similar posts which are in turn associated with users. All similar posts in the forum resulting from deviated discussions are clustered together using a novel similarity-based clustering algorithm, and each reply-to relationship existing between a pair of posts belonging to two different clusters is assumed to exist between the pair of clusters. The novelty of the proposed method lies in establishing cross-thread linkages using the post-similarity⁵ relationship, and generating a condensed social graph of the entire forum community. The main contributions of this paper can be summarized as follows.

- identification of implied *reply-to* relationships;
- clustering of similar posts based on content, title, author and time; and
- modeling community as a network of message clusters that can be explored at different levels of specificity.

For this, the method starts with linking posts in each thread individually by identifying reply-to relationships. Thereafter, a clustering algorithm based on similarity of posts is applied across the forum to group all posts into different clusters. Finally, each relation between two individual posts is mapped between the clusters to which these posts belong. Hence, the enriched interaction graph generates a network of clusters that can also be presented in the form of users who authored those posts to generate the social network of users, and at the same time it keeps information for all other users with similar interests. This work is an extension of [2] published in the proceedings of MSM'12.

The rest of the paper is organized as follows. Starting with a review of prior studies on interaction graph generation for Web forums in section 2, we discuss the enriched social graph generation method in section 3. Section 4 presents the experimental setup and evaluation results to establish efficacy of the proposed method including a brief discussion. Finally, section 5 concludes the paper with possible future enhancements.

2 Related Work

Irrespective of the type of social media (SNS, Web log, forum, video-sharing site, etc.) research on characterizing user behaviors and their interaction structures have always been a pioneering area in field of social media analysis [4, 25]. Generally, interaction structure among users play an important role in generating relevant social networks from their communications, which in turn can be applied to mine all other related information.

The inherent complexities and lack of support from the online platforms powering forums bring about various challenges in capturing user interaction structures, roles and behaviors. Identifying user roles is a well established problem in Web forum analysis. Himelboim *et al.* [19] analyzed social roles in political

⁵ The word “post” appearing throughout the paper refers to its noun form as “message-post”, and should not be confused with its adjective form.

forums to distinguish between social leaders and the rest. Chan and Hayes [7] established user communication roles in discussion forums by analyzing several different categories of features including structural, reciprocity, persistence, popularity, and initialization.

Gomez *et al.* [14] created a social network from discussion threads in *Slashdot* using user interactions and their main objective remained statistical analysis of the generated network. The Hybrid Interactional Coherence (HIC) algorithm [12] generates an interaction graph of users that is basically composed of reply-to interactions. As reply-to relations are not always explicit in a Web forum, Fu *et al.* adopted three key feature-matches including system feature match (consisting of header information match and quotation match), linguistic feature match (consisting of direct address match and a lexical match algorithm), and residual match. Rather than using the reply-to relationship between posts, Liu *et al.* [25] exploited the similarity measure to generate structure of the social network of a forum. They defined similar people to represent friendship, shared interest, or skill-similarity. For similarity comparison between different posts, they defined a measure that considers post content similarity, thread title similarity, and author similarity. In [22], Kang and Kim generated an information flow network from discussion threads. In their network, a node represents either a user or a message, and an edge represents the reply-to or authorship relationship. Messages posted by same user are connected globally across the forum in different threads using an authorship relationship. Unlike others, Aumayr *et al.* [3] applied a machine learning approach to capture the reply-to relationships using a set of five fundamental features as *reply distance*, *time difference*, *quotes*, *cosine similarity*, and *thread length*. They used SVM (support vector machine) and C4.5 (extension of decision tree based ID3 algorithm) classifiers and comparatively analyzed them by varying the feature combinations. In our earlier work [1], we have applied a similarity-based clustering approach to identify cliques in dark Web forums.

3 Proposed Method

The proposed enriched social graph mining method primarily consists of five major tasks as shown in Figure 2. It starts with forum crawling and parsing to fetch thread data, which is followed by some preprocessing tasks, reply-to relationship identification, similarity-based clustering, and finally, their integration to construct the enriched social graph. Further detail is presented in the following subsections.

3.1 Forum Crawling and Pre-processing

The process starts with data crawling and pre-processing step, in which a URL of the forum homepage is passed to the forum crawler⁶ which crawls all the

⁶ Our crawler is based on `crawler4j` package (<http://code.google.com/p/crawler4j/>)

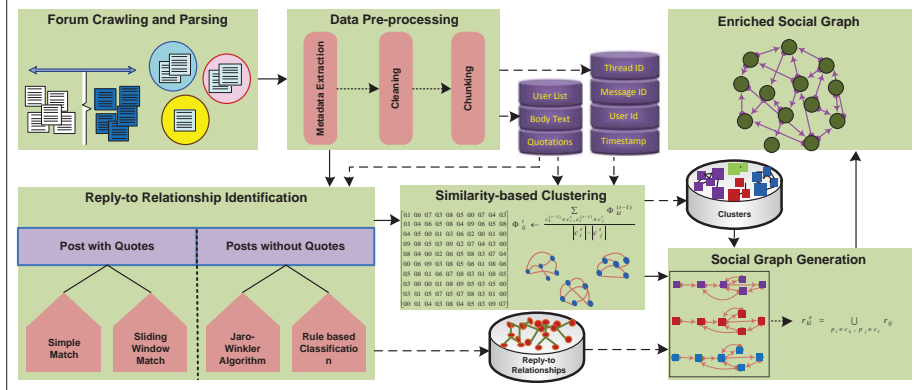


Fig. 2. Proposed social graph mining method

webpages in this domain and eliminates the duplicates heuristically. A platform-specific parser module is employed to extract all the meaningful pieces of information from the crawled webpages, which are passed to the data pre-processing module. The metadata extraction task works in close coordination with the parser module to extract all the metadata. Thus, we get the data organized as a collection of threads having a title and a unique id, each thread consisting of one or more posts that in turn comprises a post id, time-stamp, body, author and quotations, if they exist. Details about each author comprising user id, joining date, location, and total posts are collected separately. The body text is additionally processed by some cleaning and chunking to smoothen its noise and tokenize into individual meaningful pieces of information. The most common form of noise in message posts are the unnecessary repeated use of characters like punctuation marks, symbols, letters, and digits along with letters, e.g., “okkkk”. These kinds of noises are dealt by cleaning the body text. Then the body text is divided into different text chunks, called chunking, where boundaries are decided by the punctuation symbols like full-stop, comma, colon, and semicolon. It leads to produce good quality n-grams in subsection 3.3.

3.2 Reply-to Relationship Identification

When a thread is initiated by someone, it is assigned a title, and an initial post is attached with it, often called *entry post*. The entry post simply elaborates its title and waits for other’s comments on it. Viewers, who find interest on the newly initiated thread, comment on it, 1) either by quoting an existing post to respond specifically, 2) or by a quote-less post. In the first case when somebody quotes a post, the reply-to relationship becomes absolutely clear, but it is just the reverse in the other case. For example, let us suppose a user u_1 initiates a thread and another user u_2 comments on it. If u_2 commented by quoting the post of u_1 , then u_2 makes it clear that she is replying to u_1 ($u_2 \Rightarrow u_1$), but if

she commented simply without any quote, it remains unclear that to whom did u_2 reply. At the same time, as the post of u_1 is the only post other than her own, it indirectly resolves to establish the relationship as $u_2 \Rightarrow u_1$. After that, suppose u_3 replies in that thread. In case she quotes someone (either u_1 or u_2), the reply-to relationship becomes clear, else it becomes ambiguous and all the relations, $u_3 \Rightarrow u_1$, $u_3 \Rightarrow u_2$ and $u_3 \Rightarrow \{u_1, u_2\}$ have equal probability to exist. In this way, the more a thread grows in length, more ambiguous becomes the reply-to relationship. This section presents an approach to establish the reply-to relationship for each post commented in a thread.

Case 1: Posts with quotes Most of the time quotations accompanying a post occur as a simple single quote to another post. Multiple quotes (a post quoting multiple other posts at a time) and nested quotes (a post quoting a quoted post), are also encountered occasionally to focus on specific points in the discussion. All of them are neutralized by breaking down the multiple quotes into multiple single quotes, and processing the nested quotes to drop all the nested inner quotes except the outermost. Other issues regarding quotes are that sometimes a Web forum engine may itself modify the format of quotation and it also provides authority to the author to modify a default quote [28]. An author may sometimes find a lengthy quote message to be cumbersome, and to focus on a specific point may edit the message to delete rest of its body. In this kind of behavior, it becomes difficult to trace the post to which is it responding by the quote. To overcome these issues, if a simple complete match fails to identify a reply-to relation, we follow a sliding window technique [26, 12]. In this technique, the text of earlier posts as well as the quote is broken down into substrings (windows) and the quote-post pair with highest number of substring matches are linked.

Case 2: Posts without any quote For comments that are posted without quoting any of the existing posts, because of having no sound clue it becomes very difficult to establish the reply-to (\Rightarrow) relationship. Although some prior research works use the notion of similarity of posts to establish a reply-to relationship [12], contradictory to this, we found that simply a similarity of textual contents doesn't provide much evidence for a reply-to linkage. Rather a higher similarity shows an imitation of the same words. For example, let us suppose a thread is initiated with an entry post, p_1 , asking for help to learn Java, and a Java expert after noticing p_1 , replies in p_2 ($p_2 \Rightarrow p_1$) by explaining some basic concepts of Java. Another Java expert caught attention of this discussion and replied in p_3 to p_1 ($p_3 \Rightarrow p_1$) by explaining some more concepts. Now, in this thread as p_2 and p_3 are explaining on the same topic and p_1 is just a question asking help, p_1 and p_2 have a high probability to be similar even more than p_1 , but neither is replying to the other. Thus in this paper, we have differentiated a reply-to relationship from the property of posts being similar.

While commenting in a thread, very often people use author name of an earlier post in text to reply to that specific user, instead of quoting [12]. To

capture this information, a search for a match of usernames of earlier posts in the body text may lead to establish an obscured reply-to link. At the same time, as we know that an online conversation is hardly given a serious attention, the writing style remains far from a formal way of writing [18]. Unintentional misspellings and grammatical errors are commonly found in them [17], and many times usernames which do not look like real names are intentionally trimmed to make it like a real name. To overcome this hurdle, we apply an approximate string matching (ASM) algorithm. In [8], Cohen *et al.* performed a comparative study of string distance metrics for name matching and found Jaro-Winkler metric [30] as intended primarily for short strings. In our research study of Web forums, we found almost 90% of usernames to be in single words and the Jaro-Winkler metric suited best for us to match misspelled usernames.

First we define two basic measures used in it. For two strings $s_1 = a_1 \cdots a_k$ and $s_2 = b_1 \cdots b_l$, a character a_i in s_1 is defined to be *common* with s_2 if there is a $b_j = a_i$ in s_2 such that $(i - H) \leq j \leq (i + H)$, where H is calculated using equation 1.

$$H = \frac{\min(|s_1|, |s_2|)}{2} \quad (1)$$

Now, let us suppose $s'_1 = a'_1 \cdots a'_{k'}$ be the characters in s_1 which are common with s_2 (in the same order they appear in s_1) and let $s'_2 = b'_1 \cdots b'_{l'}$ be analogous to s'_1 . A *transposition* of s'_1 and s'_2 is defined to be a position i such that $a'_i \neq b'_i$. The basic Jaro metric [20] measures the similarity, $J(s_1, s_2)$, between s_1 and s_2 , using equation 2, where, $t_{(s'_1, s'_2)}$ is half the number of *transpositions* for s'_1 and s'_2 .

$$J(s_1, s_2) = \frac{1}{3} \times \left(\frac{|s'_1|}{|s_1|} + \frac{|s'_2|}{|s_2|} + \frac{|s'_1| - t_{(s'_1, s'_2)}}{|s'_1|} \right) \quad (2)$$

Based on an observation that most common typographic variations occur towards the end of a string, Winkler [30] enhanced the Jaro similarity function into equation 3, where $P' = \max(P, 4)$, P being the number of characters in the longest common prefix in s_1 and s_2 .

$$JW(s_1, s_2) = J(s_1, s_2) + \frac{P' \times (1 - J(s_1, s_2))}{10} \quad (3)$$

The value of JW metric is calculated for each pair consisting of a username from earlier posts and an n-gram in the body text. While computing the values, single word usernames are paired with uni-grams of body text, double word usernames are paired with bi-grams, and so on. A threshold value is used to confirm a match with the misspelled name, and accordingly a reply-to relationship is established with the post authored by the matched username. In case more than one post exists from that user, the relationship is linked with the latest post.

Even after applying username string matching algorithm in the body text, there remains considerable number of reply-to relationships undiscovered, and to identify which we follow a rule based classification. In this matching, we make use of communication patterns as in HIC [12], which are briefed below.

Rule Set Let x be the residual message of author A , y be the previous message of author A , and Z be the set of all the messages of other authors which are posted between y and x and are replies to messages of author A .

- Rule 1:* If y does not exist, x replies to the first message in the discussion;
- Rule 2:* If y exists and Z isn't empty, x replies to all the message posts in Z ;
- Rule 3:* If y exists and Z is empty, x replies to what y replies to.

3.3 Post Similarity Identification

Prior research show that a similarity comparison of Web forum posts is not as trivial as usual content similarity [25]. Liu *et al.* [25] defined this measure as a function of body text appended by thread title and author of the post. In our analysis, we noticed an additional factor to count for the similarity measure. Generally, time plays a substantial role in deciding the topics of discussion and its deviation, with respect to the daily happenings in one's personal life. For example, immediately after the tsunami outbreak in Japan in March 2011, all social media got flooded with this hot discussion all over the world. Hence, we observed that the discussions going in close proximity are likely to be more similar than those with a considerable time gap, and we have incorporated timestamp of a post along with other factors to measure similarity as described here.

To find overall similarity between a pair of posts, we calculate four different similarity measures as *content similarity*, *title similarity*, *author similarity* and *time similarity*. Let $D = \{d_1, d_2, \dots, d_n\}$ be the set of discussion threads and $P^i = \{p_1^i, p_2^i, \dots, p_m^i\}$ be the set of ordered posts in thread d_i in a forum F . After being cleaned and chunked in the pre-processing step, each post p_j^i is converted into bag of unigrams, bigrams and trigrams, separately. Those either beginning or ending with a stopword are filtered out. We use the vector space model (VSM) to transform each post into vectors of unigrams, \overline{Un}_j^i , bigrams, \overline{Bi}_j^i , and trigrams, \overline{Tr}_j^i , using their *tf-idf* values. The content similarity $CSim(p_j^i, p_l^k)$ between each pair of posts, p_j^i and p_l^k is calculated using equation 4, where $\alpha_1 \leq \alpha_2 \leq \alpha_3$ are constants such that $\alpha_1 + \alpha_2 + \alpha_3 = 1$.

$$\begin{aligned}
 CSim(p_j^i, p_l^k) = & \alpha_1 \times \frac{\overline{Un}_j^i \cdot \overline{Un}_l^k}{\|\overline{Un}_j^i\| \|\overline{Un}_l^k\|} + \alpha_2 \times \frac{\overline{Bi}_j^i \cdot \overline{Bi}_l^k}{\|\overline{Bi}_j^i\| \|\overline{Bi}_l^k\|} \\
 & + \alpha_3 \times \frac{\overline{Tr}_j^i \cdot \overline{Tr}_l^k}{\|\overline{Tr}_j^i\| \|\overline{Tr}_l^k\|}
 \end{aligned} \tag{4}$$

Thread title similarity, $LSim(p_j^i, p_l^k)$ is calculated in the same way as content similarity. The only difference lies in the text content which in this case is the text of thread title, as shown in equation 5.

$$LSim(p_j^i, p_l^k) = CSim(title(p_j^i), title(p_l^k)) \tag{5}$$

Author similarity, $ASim(p_j^i, p_l^k)$, is calculated using equation 6, whereas time similarity $TSim(p_j^i, p_l^k)$ is calculated using equation 7, where, $ts()$ stands for the timestamp of the associated post, and $\beta_1 \in [0, 1]$ is a constant.

$$ASim(p_j^i, p_l^k) = I_{[author(p_j^i) == author(p_l^k)]} \quad (6)$$

$$TSim(p_j^i, p_l^k) = \beta_1^{|ts(p_j^i) - ts(p_l^k)|} \quad (7)$$

Finally the overall similarity, $Sim(p_j^i, p_l^k) \in [0, 1]$, is defined by aggregating all four measures using equation 8, where α , β , γ and δ are constants such that $\alpha + \beta + \gamma + \delta = 1$.

$$\begin{aligned} Sim(p_j^i, p_l^k) &= \alpha \times CSim(p_j^i, p_l^k) + \beta \times TSim(p_j^i, p_l^k) \\ &+ \gamma \times ASim(p_j^i, p_l^k) + \delta \times LSim(p_j^i, p_l^k) \end{aligned} \quad (8)$$

3.4 Thread Post Clustering

Online threaded discussions usually start from a specific topic but as a thread grows with more posts, it's context deviates from its actual title [15]. Very often this deviated discussion is found to be overlapping with another one going on in a different thread. To capture this inter-thread similarity, in this step we follow a cost-effective agglomerative clustering algorithm shown in Algorithm 1 to group all similar posts across the forum. It starts with assigning all the different forum posts in a separate cluster. Let us suppose there are n_0 number of total posts in the forum and at time $t = 0$ it starts with $C^0 = \{c_1^0, c_2^0, \dots, c_{n_0}^0\}$ as the set of clusters assuming that every post is dissimilar from others. At each iteration, t , in the clustering process, a similarity matrix $\Phi_{n_t \times n_t}^t$ is maintained containing the similarity information between each pair of clusters. For the initial similarity matrix, $\Phi_{n_0 \times n_0}^0$, at $t = 0$ its values are calculated as a similarity measure between each pair of posts as shown in equation 9, where $p_i \in c_i^0$ and $p_j \in c_j^0$.

$$\Phi_{ij}^0 = Sim(p_i, p_j) \quad (9)$$

At time, t , each value in the matrix, $\Phi_{n_t \times n_t}^t$, is compared with the similarity threshold value, ϵ . The pair of clusters for whom this value is found to be greater are added to the set of pairs, A^t , that need to be merged. After collecting all the cluster pairs that show a sign to get merged, they are ranked by their corresponding matrix values. Starting with the top ranking pair, the two clusters are merged to form a unified cluster and all those pairs in A^t containing either of the two sub-clusters are removed from the set. The merging process is continued until A^t becomes empty. After the completion of merging, it proceeds to next iteration, $t + 1$, the new set of clusters becomes $C^{(t+1)}$ with number of clusters as $n_{(t+1)} < n_t$, and the new matrix becomes $\Phi_{n_{(t+1)} \times n_{(t+1)}}^{(t+1)}$.

Each cluster, c_i^t , at time, t , keeps information about all its posts grouped into two sub-clusters, $c_k^{(t-1)}$ and $c_l^{(t-1)}$, if c_i^t is a result of merging $c_k^{(t-1)}$ and

$c_l^{(t-1)}$, else c_i^t contains a single cluster of posts, $c_k^{(t-1)}$, the same as it was in last iteration. Each value, Φ_{ij}^t , in the new matrix is calculated using equation 10, where $|c_i^t|$ and $|c_j^t|$ denote the number of sub-clusters in c_i^t and c_j^t , respectively.

$$\Phi_{ij}^t = \frac{\sum_{c_k^{(t-1)} \in c_i^t, c_l^{(t-1)} \in c_j^t} \Phi_{kl}^{(t-1)}}{|c_i^t| \cdot |c_j^t|} \quad (10)$$

Algorithm 1: Post clustering algorithm

Input: A set of posts $P = \{p_1, p_2, \dots, p_n\}$
Output: A set of cluster of posts $C = \{c_1, c_2, \dots, c_m\}$

- 1 $C^0 = \{c_1^0 \leftarrow p_1, c_2^0 \leftarrow p_2, \dots, c_{n_0}^0 \leftarrow p_n\};$
- 2 $t \leftarrow 0;$
- 3 **repeat**
- 4 **if** $t = 0$ **then**
- 5 $\Phi_{n_t \times n_t}^t \leftarrow \text{createSimilarityMatrix}(C^0);$
- 6 **else**
- 7 $\Phi_{n_t \times n_t}^t \leftarrow \text{createMatrix}(n_t \times n_t);$
- 8 **forall the** i **and** j **in** Φ_{ij}^t **do**
- 9 $\Phi_{ij}^t = \frac{\sum_{c_k^{(t-1)} \in c_i^t, c_l^{(t-1)} \in c_j^t} \Phi_{kl}^{(t-1)}}{|c_i^t| \cdot |c_j^t|};$
- 10 **if** $\Phi_{ij}^t \geq \epsilon$ **then**
- 11 $\Lambda^t \leftarrow \Lambda^t \cup \{(c_i^t, c_j^t, \Phi_{ij}^t)\};$
- 12 $\text{rank}(\Lambda^t)$ on decreasing value of Φ_{ij}^t ;
- 13 **repeat**
- 14 $\{(c_i^t, c_j^t, \Phi_{ij}^t)\} \leftarrow \text{top}(\Lambda^t);$
- 15 $\text{merge}(c_i^t, c_j^t);$
- 16 **forall the element** $\{(c_k^t, c_l^t, \Phi_{kl}^t)\} \in \Lambda^t$ **do**
- 17 **if** $c_i^t = c_k^t$ **or** $c_i^t = c_l^t$ **or** $c_j^t = c_k^t$ **or** $c_j^t = c_l^t$ **then**
- 18 $\text{remove}\{(c_k^t, c_l^t, \Phi_{kl}^t)\}$ **from** Λ^t ;
- 19 **until** Λ^t **becomes empty**;
- 20 $t \leftarrow t + 1;$
- 21 $C^t \leftarrow \text{getClusters}();$
- 22 **until** $|C^t| = |C^{(t-1)}|;$
- 23 $C = C^t;$
- 24 **return** C

After t iterations, when there remains no Φ_{ij}^t value greater than the ϵ , the terminating condition in the algorithm shown in Algorithm 1 becomes true and the final clusters are returned as grouped posts. Some spectacular properties of the proposed clustering algorithm are presented below.

- In this algorithm, we do not need to have pre-decided number of clusters (to be generated finally) as is required in most [31]. Rather this number is determined dynamically by comparing Φ_{ij}^t with $\epsilon \in [0, 1]$.
- A strict hierarchical clustering algorithm suffers from its inability to perform adjustment once a merge or split decision has been taken [16], whereas the proposed algorithm is free from this limitation as before merging we rank the cluster pairs to make sure that the merged cluster would not need to be split up later.
- Due to heavy computations, the cost of clustering usually remains very high [31, 16]. The time complexity of the proposed clustering algorithm is tn^2 , t being the number of iterations, and in worst case it may go up to n^3 . However, during its execution as more and more sub-clusters get merged in successive iterations, the dimension of the similarity matrix decreases and the number of computations reduce heavily to make it an efficient algorithm.

3.5 Enriched Social Graph Generation and the Ecosystem Dynamics

In prior research [14, 12, 25, 3], user interactions in Web forums have been defined to generate a network of users for analyzing their activities in different ways, and the reply-to relationships are identified as most prominent features to trace them. In addition, post similarity has been found as another important feature to define a network of users of similar interests [25]. In HIC [12], similarity among the posts in a thread are used as a heuristic to establish a reply-to relationship. However the proposed web forum ecosystem modeling method differentiates a reply-to relationship from the property of posts being similar. The enriched social graph considers a cluster of similar posts in the forum as a node, and the reply-to relationships between posts from different clusters as directed links to connect the nodes. Let $P = \{p_1, p_2, \dots, p_m\}$ be the set of total posts in all the threads and $R = \{r_{ij}\}$ be the set of relationships $p_i \Rightarrow p_j$ between posts, p_i and p_j . Let us suppose that the set of clusters generated using the clustering algorithm is $C = \{c_1, c_2, \dots, c_n\}$. Now, the enriched social graph consists of n cluster nodes with the set of relationships, $R^e = \{r_{kl}^e\}$, where r_{kl}^e is defined in equation 11.

$$r_{kl}^e = \bigcup_{p_i \in c_k, p_j \in c_l} r_{ij} \quad (11)$$

Each post associates with it the thread title, author name or user id and timestamp, and this enriched social graph can be presented in various forms for analyzing the interactions and associations in the ecosystem. When the graph is presented for authors of the posts, it generates a network of authors resembling the *ecosystem actors* connected to others by the reply-to relationship, and at the same time shows all existing actors with similar interest being in the same cluster. An actor with diverse interests may exist in multiple clusters, which shows the diverse nature of her interests. When the clusters are represented by keywords of posts resembling the *ecosystem environment components*, a relation

between the topics on discussion issues (or ecosystem environment components) can be identified. When clusters are represented by the timestamp along with keywords of posts, a contemporary network of the hot discussion topics can be generated which characterizes the *ecosystem evolution*. Hence, the generated enriched social graph is actually a multi-purpose graph that captures the distinct features and interactions of a Web forum ecosystem and can be very fruitful for an exhaustive analysis.

4 Experimental Results

The experiments are conducted on a real time test bed described in subsection 4.1. Firstly, the crawler module based on `crawler4j` and parser module developed for the `vBulletin` platform crawls and parses the forum webpages to extract all the meaningful pieces of information from them, which are then passed on to subsequent modules. The proposed social graph generation method is evaluated in different perspectives in subsection 4.2.

4.1 Test Bed

For a realtime test bed, we considered “*eActivism and Stormfront Webmasters*” forum under the *Activism* category in the popular Stormfront⁷ social Web forum. As of 04th Feb. 2012, this sub-forum consisted of 1,698 threads getting a total of 11,210 replies and 2,271,494 views. Set up in 1995 by Don Black, it is considered by many as a neo-Nazi Web forum and was identified as the first major hate-site on the Web, which drove us to study user interactions in it.

4.2 Experiments

The first major task is to identify the reply-to relationships in-between posts in a thread, that lay a foundation of the proposed graph generation method. This task is evaluated by using the metrics, precision (π), recall (ρ) and F-score (F_1). Precision is defined as the ratio of no. of correctly identified relations to total no. of identified relations, recall is defined as the ratio of no. of correctly identified relations to the no. of relations that actually exist, and F-score is harmonic mean of the duo.

As the metrics needed a gold standard to compare with, some agelong threads relevant to the forum category are shortlisted to manually set its gold standard. Only 29 threads are found having more than 40 comments on them. Discarding the irrelevant ones, we stuck to 10 threads. Two independent users are assigned to manually identify all actual reply-to relationships based on their context, and finally conflicts were resolved on a mutual consent. Another set of relationships identified by the proposed method are also collected. Values of the evaluation metrics are calculated using these two sets, shown in Table 1 along with their

⁷ <http://www.stormfront.org>

statistical summary. In these 10 threads, we see that the F-score value reached as high as 0.900 for thread no. 10 and as low as 0.711 for thread no. 6. The average values for precision, recall and F-score are found as 0.799, 0.815 and 0.806 respectively. Out of the two cases mentioned in subsection 3.2 to capture the relationship, the first case was found to be accurate with 0.987 as its average F-score. However in the second case, due to its high ambiguity, accuracy of the system went down to an average F-score of 0.641.

Table 1. Result summary of *reply-to* relationship identification process

Thread No.	Posts	Participants	π	ρ	F_1
1	68	22	0.784	0.816	0.800
2	105	13	0.727	0.715	0.721
3	122	37	0.831	0.847	0.839
4	82	54	0.802	0.790	0.796
5	185	48	0.878	0.845	0.861
6	58	41	0.691	0.733	0.711
7	55	11	0.856	0.862	0.859
8	169	52	0.773	0.816	0.794
9	44	11	0.758	0.809	0.783
10	46	37	0.887	0.914	0.900
Avg.	93.4	32.6	0.799	0.815	0.806

As another part of this experiment, the established relationships between posts are transformed to establish the same relations between users. As there exist some common users in different threads, the relationship established for a user in one thread is continued over and integrated with the relationships in other threads, which connected users in different threads to form a network. The generated network consisted of 3 distinct components, each of whom represent a closed-group of inter-related users, shown in Figure 3. There are a total of 310 nodes (overall participants) and 545 directed edges (reply-to relationships) generated from a total of 934 posts distributed in 10 discussion threads. Figure 4 presents a zoomed view of the two small components.

In the next experiment, we considered to test the clustering algorithm based on the designed post similarity measure. It is evaluated in terms of the metrics $F_{\alpha=0.5}$ (or F_P ⁸ at $\alpha = 0.5$) and $F_{B-cubed}$ (or F_B ⁹). As like the previous experiment, posts are manually grouped by two independent users and conflicts are resolved on mutual consent, which produced a set of 207 clusters as a gold

⁸ If C is the set of clusters generated by the automated system and L is the gold standard set, then $purity = \sum_i \frac{|C_i|}{n} \max Precision(C_i, L_j)$ and $inversepurity = \sum_i \frac{|L_i|}{n} \max Precision(L_i, C_j)$. F_P is calculated as their harmonic mean.

⁹ For each element (or post), i , precision and recall values are computed individually as $precision_i = \frac{C_i \cap L_i}{C_i}$ and $recall_i = \frac{C_i \cap L_i}{L_i}$. The average b-cubed precision and recall are computed as the mean of individual values. F_B is the calculated as their harmonic mean.

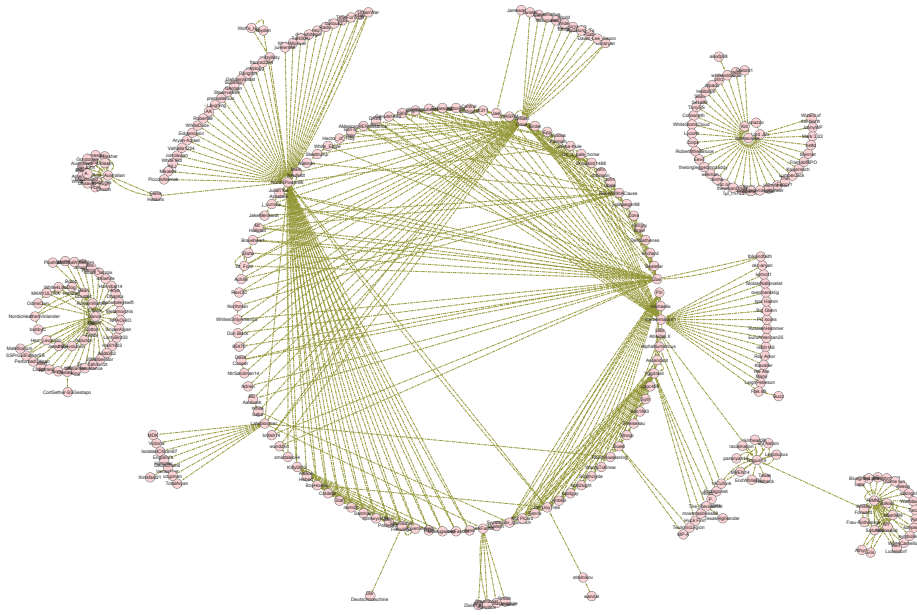


Fig. 3. Network generated by system-identified *reply-to* relations between users

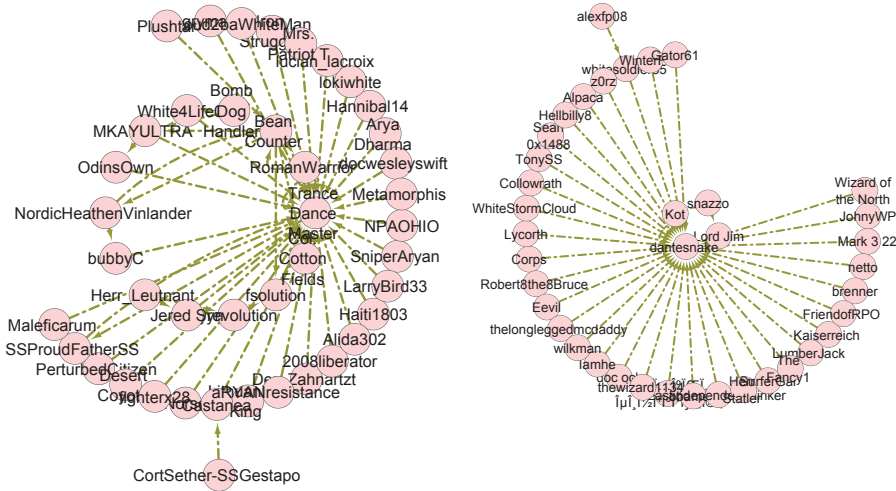


Fig. 4. Zoomed view of the two small components in Figure 3

standard from the same set of 934 posts. The proposed method is started with extracting all the unigrams, bigrams, and trigrams. For the constants, α_1 , α_2 and α_3 , their values are determined by solving the equations, $\alpha_3 = 3 \times \alpha_1$, and $\alpha_2 = 2 \times \alpha_1$, where $\alpha_1 + \alpha_2 + \alpha_3 = 1$ is the condition applied. Thus their values are computed as 0.167, 0.333 and 0.5 for α_1 , α_2 and α_3 respectively. The interaction style in Web forums is not of instant nature and many times the lifetime of a thread even go to a year or more. Therefore, for computing time similarity, the time difference is calculated in unit of hours, and value of β_1 is experimentally set to 0.995. Tuning the parameters α , β , γ and δ , is another challenge. The ideal way is to learn them from the manually annotated set, and we leave it as an application issue. In our case, we experimentally set them to 0.7, 0.1, 0.1 and 0.1, respectively, and generate the similarity matrix. Thereafter the clustering algorithm is executed by varying similarity threshold, ϵ , from 0.2 to 0.5 in intervals of 0.05. The line chart shown in Figure 5 presents the trend of increasing number of generated clusters as the value of ϵ increases. As we move away from the value $\epsilon = 0.3$ in either side, the difference between the number of automatically generated and manually identified clusters goes on broadening, which leads to a fall in accuracy of the algorithm. Figure 6 presents the

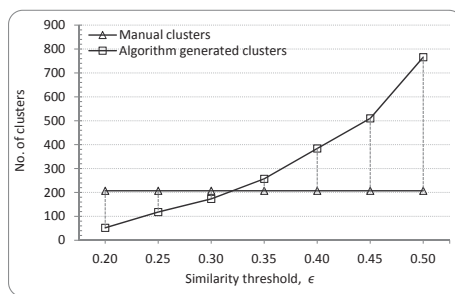


Fig. 5. A comparison of automatically generated clusters

impact of varying ϵ on the evaluation metrics. The purity and average b-cubed precision values decrease to 50.7% and 37.7% respectively at $\epsilon = 0.2$, and the inverse purity and average b-cubed recall values increase to 86.4% and 91.4% respectively. On the other end, at $\epsilon = 0.5$ the values go to 90.5% and 85.9% respectively for purity and average b-cubed precision, and 44.7% and 33.9% for inverse purity and average b-cubed recall. Accordingly reflections are shown in F_P and F_B measures. Considering $\epsilon = 0.3$ as the ideal threshold, the F_P and F_B values in this experiment are found as 0.825 and 0.804, respectively. A detailed result summary is presented in Table 2.

Thereafter, proceeding forth with $\epsilon = 0.3$, all the 545 reply-to relationships among 934 posts are unified to construct the social graph at cluster-level. On unifying these post-to-post relations to map them to cluster-to-cluster relations,

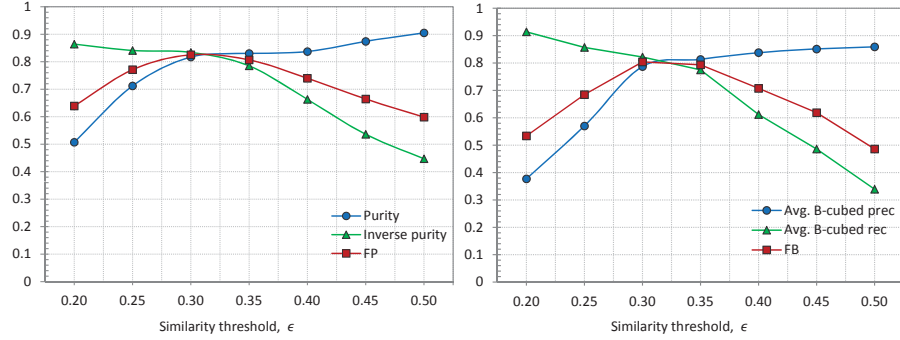


Fig. 6. Impact of ϵ on *Purity* measures and *B-Cubed* measures

Table 2. Result summary of similarity-based clustering algorithm

Metric	Value	Metric	Value
Purity	0.817	Avg. B-cubed Prec	0.787
Inverse Purity	0.834	Avg. B-cubed Rec	0.822
F_P	0.825	F_B	0.804

we got 332 relations in-between 173 clusters, identified above. Figure 7 visualizes the generated enriched social graph. To keep it simple and easy for visual perception, the internal informative details have not been displayed. Each node in it is a cluster of posts that are highly similar to each other and each link is a directed reply-to relationship between the two clusters. We see that it consists of two disconnected components, which shows that the posts in one component is neither similar to those in the other, nor the posts in one component replied to any post of the other component. Thus, the small component is either a single thread or a group of very few threads whose topic of discussion is totally different from that going in rest of the threads. As can be seen in the larger component that few nodes are thickly connected to others while most are very thinly connected. The set of posts in thickly connected nodes are getting more attention from other members for the inbound links and their outbound links show that their authors are active members in the forum. Irrespective of being inbound or outbound, the thickness of linkages indicates that the topics of posts in the cluster are among the hot issues of that time. The constructed enriched social graph can be used to present the network in multiple ways as described in section 3.5 to present and analyze the dynamics of the web forum ecosystem. The thickness of linkages in the social graph characterizes the kind of ecosystem. A social graph with thick inter-cluster collaborations and linkages create a dense network that characterizes a *strong* ecosystem, whereas a social graph with thin inter-cluster linkages create a sparse network that characterizes a *weak* ecosystem.

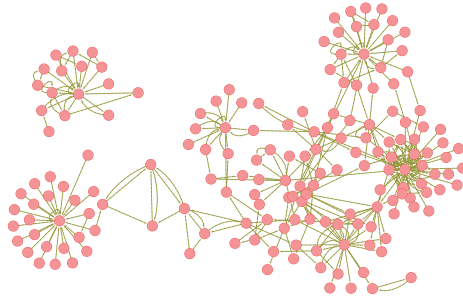


Fig. 7. Generated social graph

Although no experiment is performed on Q/A Web forums such as *cross-validated*¹⁰ or *stack-overflow*¹¹, the methodology is highly applicable to them and such other specific forums. It would start with mining the enriched social graph from the complete Q/A forum and keep on updating the graph at regular intervals with the addition of threads and posts in the forum. It could then be used for automatic spontaneous query-answering using the knowledge stored in the graph.

5 Conclusion and Future Work

In this paper, we have considered a Web forum as an ecosystem and presented a methodology to model the discussions into an enriched social graph using user interactions and their overlapping interests, with a deliberate consideration of deviated discussions. The user interactions link posts through *reply-to* relationships, whereas the overlapping interests lead to merge similar posts into clusters, and thus collapse the generated network. Authors of posts in the same cluster share common interests and are linked with people of distinctive interests through the *reply-to* tie. The enriched social graph can serve for analysis of Web forum discourse in multiple ways that can be explored to bring various undiscovered facts. It is also applicable to Q/A Web forums and such other specific forums for automatic query-answering spontaneously.

This work mainly focuses on the approach to generate social graph to model user interactions and their overlapping interests, rather than analyzing forums using it. Therefore, the most important future direction is to devise approaches to analyze Web forums using the generated social graph. There are few issues regarding enhancements of the proposed approach. The linguistic analysis of posts can further be enriched to improve the F-score value of *reply-to* relation identification process for the posts which do not include quotes. Moreover, usually people in real life are tied together by several other kinds of social relationships,

¹⁰ <http://crossvalidated.com/>

¹¹ <http://stackoverflow.com/>

which somewhat also exist on the Web. Identification and incorporation of all such relationships in the social graph, along with user activities and behaviors, are good candidates for future work.

References

1. T. Anwar and M. Abulaish. Identifying cliques in dark web forums- an agglomerative clustering approach. In *Proc. of the 10th IEEE Int'l Conf. on ISI*, pages 171–173, 2012.
2. T. Anwar and M. Abulaish. Mining an enriched social graph to model cross-thread community interactions and interests. In *Proc. of the 3th Int'l Workshop on MSM, Co-located with 23rd ACM Int'l Conf. on HT*, pages 35–38, 2012.
3. E. Aumayr, J. Chan, and C. Hayes. Reconstruction of threaded conversations in online discussion forums. In *Proc. of the AAAI ICWSM*, pages 26–33, 2011.
4. F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida. Characterizing user behavior in online social networks. In *Proc. of the 9th ACM SIGCOMM Internet Measurement Conf.*, pages 49–62, 2009.
5. C. A. Bentivoglio. Recognizing community interaction states in discussion forum evolution. In *AAAI Fall Symposium Series*, pages 20–25, 2009.
6. B. E. Brewington and G. Cybenko. How dynamic is the web? *Comput. Netw.*, 33(1-6):257–276, 2000.
7. J. Chan, C. Hayes, and E. Daly. Decomposing Discussion Forums using User Roles. In *Proc. of the WebSci10: Extending the Frontiers of Society On-Line*, 2010.
8. W. Cohen, P. Ravikumar, and S. Fienberg. A comparison of string distance metrics for name-matching tasks. In *Proc. of the Int'l Workshop on IIWeb, held with IJCAI*, pages 73–78, 2003.
9. T. Correa, A. W. Hinsley, and H. G. de Zúñiga. Who interacts on the web?: The intersection of users' personality and social media use. *Comput. Hum. Behav.*, 26(2):247–253, 2010.
10. M. De Choudhury, W. A. Mason, J. M. Hofman, and D. J. Watts. Inferring relevant social networks from interpersonal communication. In *Proc. of the 19th Int'l Conf. on WWW*, pages 301–310, 2010.
11. A. El Abaddi, L. Backstrom, S. Chakrabarti, A. Jaimes, J. Leskovec, and A. Tomkins. Social media: source of information or bunch of noise. In *Proc. of the 20th Int'l Conf. Companion on WWW*, pages 327–328, 2011.
12. T. Fu, A. Abbasi, and H. Chen. A hybrid approach to web forum interactional coherence analysis. *J. Am. Soc. Inf. Sci. Technol.*, 59(8):1195–1209, 2008.
13. E. Gilbert and K. Karahalios. Predicting tie strength with social media. In *Proc. of the 27th Int'l Conf. on Human Fact. in Comp. Sys.*, pages 211–220, 2009.
14. V. Gómez, A. Kaltenbrunner, and V. López. Statistical analysis of the social network and discussion threads in slashdot. In *Proc. of the Int'l Conf. on WWW*, pages 645–654, 2008.
15. Y.-H. Guan, C.-C. Tsai, and F.-K. Hwang. Content analysis of online discussion on a senior-high-school discussion forum of a virtual physics laboratory. *Instructional Science*, 34(4):279–311, 2006.
16. J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*, pages 408–418. Morgan Kaufmann, 2 edition, 2006.
17. E. Hargittai. Hurdles to Information Seeking: Spelling and Typographical Mistakes During Users' Online Behavior. *J. of the Assoc. for Information Systems*, 7(1):52–67, 2006.

18. S. C. Herring. Computer-mediated communication on the internet. *Ann. Rev. of Inf. Sc. and Tech.*, 36(1):109–168, 2002.
19. I. Himelboim, E. Gleave, and M. Smith. Discussion catalysts in online political discussions: Content importers and conversation starters. *J. of Computer-Mediated Comm.*, 14(4):771–789, 2009.
20. M. A. Jaro. Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. *J. of the Am. Statistical Assoc.*, 84(406):414–420, 1989.
21. S. Jones and S. Fox. Generations online in 2009. Technical report, PewResearch Center, 2009. <http://www.pewinternet.org/Reports/2009/Generations-Online-in-2009.aspx>.
22. J.-H. Kang and J. Kim. Analyzing answers in threaded discussions using a role-based information network. In *Proc. of the 3rd IEEE Int'l Conf. on Soc. Comp.*, 2011.
23. A. M. Kaplan and M. Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business Horizons*, 53(1):59–68, 2010.
24. A. Lenhart. Adults and social network websites. Technical report, PewResearch Center, 2009. <http://www.pewinternet.org/Reports/2009/Adults-and-Social-Network-Websites.aspx>.
25. D. Liu, D. Percival, and S. E. Fienberg. User interest and interaction structure in online forums. In *Proc. of the 4th Int'l AAAI Conf. on Weblogs and Soc. Med.*, pages 283–286, 2010.
26. T. Nahnsen, O. Uzuner, and B. Katz. Lexical chains and sliding locality windows in content-based text similarity detection. Technical report, MIT (CSAIL), 2005. MIT-CSAIL-TR-2005-034, AIM-2005-017, <http://dspace.mit.edu/handle/1721.1/30546>.
27. C. P. Rosé, B. Di Eugenio, L. S. Levin, and Carol. Discourse processing of dialogues with multiple threads. In *Proc. of the 33rd Ann. Meet. on Assoc. for Comp. Ling.*, pages 31–38, 1995.
28. K. Severinson Eklundh. To quote or not to quote: Setting the context for computer-mediated dialogues. *Language@Internet*, 7(5), 2010.
29. J. van Dijck. Users like you? theorizing agency in user-generated content. *Media Culture Society*, 31(1):41–58, 2009.
30. W. E. Winkler. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage. In *Proc. of the Section on Survey Research*, pages 354–359, 1990.
31. R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Trans. on Neural Networks*, 16(3):645–678, 2005.