Clustering and Classification of Like-Minded People from Their Tweets

Soufiene Jaffali, Salma Jamoussi, Ben Abdelmajid, Kamel Smaili

Clustering and Classification of Like-Minded People from Their Tweets

Soufiene Jaffali, Salma Jamoussi, Abdelmajid Ben Hamadou
MIRACL Laboratory
University of Sfax
Sfax, Tunisia
{*jaffali.soufiene, jamoussi, abdelmajid.benhamadou*}*@gmail.com*

Kamel Smaili
SMART team
LORIA
Nancy, FRANCE
kamel.smaili@loria.fr

***Abstract*—Several challenges accompanied the growth of online social networks, such as grouping people with similar interest. Grouping like-minded people is of a high importance. Indeed, it leads to many applications like link prediction and friend or product suggestion, and explains various social phenomenon. In this paper, we present two methods of grouping like-minded people based on their textual posts. Compared to three baseline methods K-Means, LDA and the Scalable Multi-stage Clustering algorithm (SMSC), our algorithms achieves relative improvements on two corpora of tweets.**

***Keywords*-social network; like-minded users; communities discovery; text mining**

I. Introduction

A community in a social network can be defined as a set of users having similar criteria like location or political party, etc. [2], or as a group of users who share the same interests [29]. Studying communities in social networks is of growing importance. Indeed, analyzing such groups leads to build patterns and understand the evolution of social networks. Many studies deal with this subject, and different approaches are employed [19]. Also, many aspects are considered to extract communities. Most of works use explicitly friendship information or interactions to discover communities in social network [16], [18]. According to [14], the number of links by user in social networks follows a big tail distribution. Which means that just a few users have a big number of links, when, most of social network users have only few links. Thus, mining only explicit relations within the network does not provide a complete vision.

Some works aim to connect users according to their interest centers [29]. Grouping users with similar interest gives a better vision and leads to many applications such as friend suggestion, collaborative filtering, etc. In fact, suggest friends based on link information may recommend people that you already know. However, using interest centers ensure user that he will be connected to someone like him. Also, in question/answering social networks like Yahoo Answers [1], connecting like-minded users may facilitate and accelerate solving common problems.

In the present work, we propose two algorithms of grouping like-minded users. The principal idea of the first proposed algorithm is to retrieve the interest centers from the users' textual posts, and then, to group users having same interests. The second algorithm aims to retrieve groups with maximum correlation between users using the Principal Component Analysis (PCA). After creating user communities, in both methods, we use an SVM classifier to classify new users.

The rest of the paper is organized as follows. Section 2 presents the related works. Section 3 presents the proposed algorithms. We present the results in Section 4. We end with conclusions and avenues for future work in Section 5.

[1] https://answers.yahoo.com/

II. Related Work

The most of works in social network community detection are based on link information [4]. Moreover, many kinds of information are used to retrieve significant communities. Tags are deeply used to construct the user's profiles [5], and to classify the interest centers [9]. Some works create tag communities using PCA, and assign the users to the closest communities [1]. Others connect the like-minded users using the tag network inference [29]. We can find works using other information, such as the mutual awareness [10], comments and like actions [18].

Given the fact that some users do not employ tags in their posts and that the same subject can be described by more than one tag, the use of tags for community detection may not succeed or yield to unoptimized results. Therefore, we suggest retrieving the latent interest centers from textual posts, and then, using the retrieved centers in order to group the users into communities. In the literature, just a few works deal with extracting social relations between individuals from text [13]. In this context, the Latent Dirichlet Allocation (LDA) and the probabilistic Latent Semantic Analysis (pLSA) are largely implemented to generate the subject models being used to regroup the tweets [6]. Similarly, Sachan et al. [24] applies LDA to identify the subjects of discussion based on the interactions between the users. These subjects are used to create the communities in a second stage. Using LDA, Hannachi et al. [6] extract the subjects from the published tweets to build a model directed by them. In the context of social media recommendation, Pennacchiotti and Gurumurthy [21] build an LDA model

from users' tweets to discover users' interest automatically. To discover users' topics of interest, Michelson and Macskassy [15] use Wikipedia[2] to retrieve categories from users' tweets, and then, build user profile from categories.
Tsur et al. [27] propose the scalable multi-stage clustering algorithm (SMSC) in order to cluster tweets. The SMSC algorithm had been tested on a collection of tweets and presented high performances.
Compared to our previous work [8], the two novel algorithms proposed in this work had three principal contributions: (i) improve previous grouping results, (ii) take into account overlaps between groups, and (iii) classify new users into created groups.

III. Grouping Like-Minded People

A. GLIC

The main idea of our algorithm GLIC (Grouping Like-minded people using Interest Centers) is to group like-minded user, based on their publications. Thus, we extract interest centers from users' textual posts, and gather users according to the extracted centers. And so, we classify users according to their posts. Once user groups are created, we use the publication flow in order to classify the new users. In literature, only few works use interest centers extracted from textual content to find communities in social networks [6]. Those works use LDA or LSA to find topic models. In GLIC, we implement the PCA to retrieve the latent interest centers from textual posts. Figure 1 illustrates the proposed algorithm. Our algorithm consists of five principal steps detailed below.

1) Text Preprocessing: in the present work, we deal with the Twitter text messages known as "tweets". The latter are limited to 140 characters allowing users to share their status. Textual publications in social networks are neither structured nor written in a formal language which may make their exploitation very difficult. To deal with this problem, we, first, eliminate the stop words (personal pronouns, prepositions, etc.). Then, we convert all upper-case letters to lower-case ones. Next, we eliminate the words occurring less than a prefixed threshold. We experimented with various values of threshold. Using the obtained words, we build the occurrence matrix M. Where, each line (respectively column) represents a user's tweet (respectively a word from the corpus of tweets).

$$M = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{11} & \cdots & x_{2m} \\ \vdots & \ddots & x_{ij} & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{pmatrix}$$

x_{ij} is the occurrence of the word j in the user's tweet i.
Finally, we normalize the values of the matrix M, in order to obtain a normalized matrix M'. Thus, we use two different methods. The first consists on normalizing each occurrence of a word by the sum of each line:

$$x_{ij} = \frac{x_{ij}}{\sum_k x_{ik}} \quad (1)$$

In the second normalization method, we divide each value by the standard deviation of each column.

$$x_{ij} = \frac{x_{ij}}{\sigma_{x_{kj}}} \ ; \ with \ k = 1..n \quad (2)$$

With n is the number of tweets in the corpus.

2) Seeking the Latent Centers of Interest: in this step, we extract the interest centers from the tweet collection using the PCA [20]. PCA is generally used to reduce the data space or to find the axes where the data are concentrated as it the case in our study. In [7], the authors prove that the PCA extract successfully the common themes from text documents. In this part, we use PCA to determine the axes around which the words used in publications are concentrated. In other word, we retrieve the common themes or subjects from the tweet collection. Those axes are the latent interest centers within the input data. One interest center can reflect one more subjects evoked by a group of users.
To retrieve the interest centers, we calculate the covariance matrix of M' and its eigenvectors. The obtained eigenvectors present the latent interest centers within the users posts. Each interest center is of the form $C_j = \{c_{j1}, c_{j2}, .., c_{jl}, .., c_{jm}\}$ with c_{jl} is the weight of the term l in the component j. To avoid any loss of information, we maintain all the eigenvectors. Thus, we get m eigenvectors.

3) Projecting Data in the Interest Centers Space: after calculating the axes where the data are concentrated, we project our data from the original space to the interest centers space. Thus, we multiply the matrix M' by C. Where C is the matrix of interest centers, where each column is an eigenvector. We obtain a matrix M'', which is the new representation of the input data. In this matrix, tweets are represented by their coordinates in the interest centers space, instead of word occurrences. In the new space, a tweet is closer from those sharing similar interest centers with it. This representation is more adequate for the next step. Where, we aim to group tweets which having similare interest centers using the K-Means algorithm. And such grouping is based on the distances between tweets.

4) Assigning the Users to the Clusters: In this step, we employ the K-Means algorithm to regroup the like-minded users in categories. Indeed, K-Means is a grouping algorithm which classifies the objects in a number K of groups by taking into account the attribute values. In this work, we use the matrix M'' as input of K-Means. So, the tweets are clustered according to their coordinates in the interest centers space.
After building clusters of tweets, we replace each tweet by its publisher. Therefore, we obtain groups of like-minded

[2]https://www.wikipedia.org

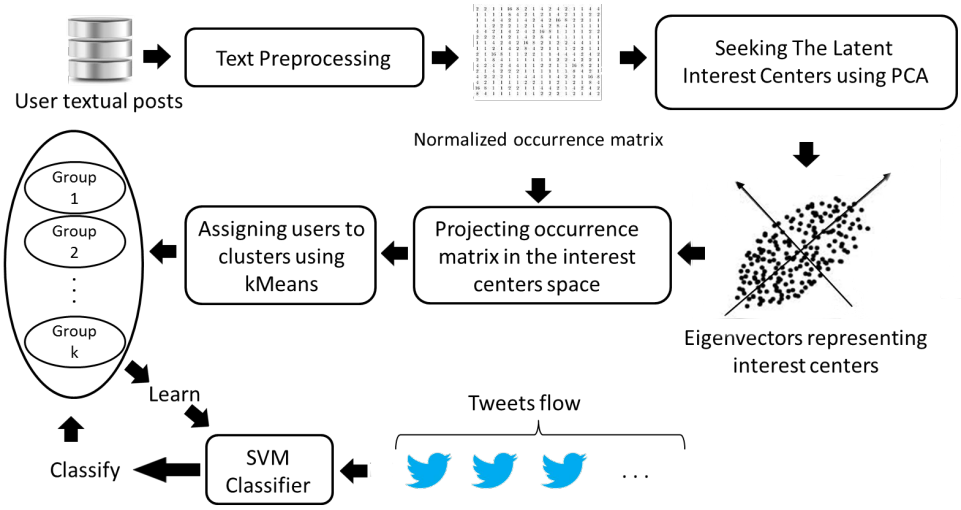

Figure 1. Proposed Process for Clustering and Classifying Like-Minded People

users instead of groups of tweets. Such clustering allows us to assign users having more than one interest center, to different groups, in opposite to what have done by [6], [27]

5) Classifying New Users: In this stage, we use the like-minded users' groups as training examples for the Support Vector Machines (SVM) algorithm [28]. Several recent studies have reported that the SVM generally are capable of delivering higher performance in terms of classification accuracy than the other data classification algorithms [26]. For training and classification using SVM, we employ the $SVMmulticlass$[3]. After training SVM, we obtain a model allowing to classify new users. We can repeat the steps (2), (3) and (4) from time to time in order to increase accuracy and reflect newly introduced interest centers.

B. GLUCA

In this method, we adopt a different approach to group like-minded users based on their textual publications. The core idea is to retrieve the user concentration axes instead of words concentration axes. In GLIC method, we use the matrix $user \times word$ to find axes concentration of the words used in tweets, these axes represent the space of interest centers in which we project the input data. In this method GLUCA (Grouping Like-minded people based on Users Concentration Axes), we use the transpose of $user \times word$ matrix as input to the PCA. Thus, we obtain the user's concentration axes instead of word's concentration axes. Once the axes are retrieved, we assign users belonging to the same axe to the same community. Hereafter, we describe the main steps of GLUCA.

1) Text Preprocessing: in this step, we proceed in the same way as in the first method to preprocessing users textual posts (described in the section III-A1). Once we obtain the occurrence matrix M, we calculate its transpose M^T. Where, each line (respectively column) represents a word of the corpus tweet (respectively a user's tweet). Using the transpose matrix M^T, we retrieve users concentration axes instead of words concentration axes.

2) Extracting User Communities Using PCA: unlike GLIC, where we use the PCA to seek latent interest centers within the tweets collection, we apply the PCA to retrieve user concentration axes. Using M^T as input, the principal components represent the user groups. Each component is a vector representing the user weights relative to the corresponding group. After retrieving the principal components, we keep those corresponding to the highest eigenvalues. Finally, we assign each user to the group that has the highest weight.

3) Classifying New Users: to classify the flow of new users, we proceed in the same manner as in the first method. We use the user groups found in the previous step to learn the SVM classifier. Then, we utilize this later to classify new users.

IV. Experimentations

A. Baseline

To evaluate our algorithms and to compare them to the works of the literature, we use K-Means, LDA and SMSC as reference grouping algorithms.

1) K-Means: As a first reference, we use the classical K-Means [12], one of the most used algorithms for the clustering. [11] test three k-means methods based on optimal prediction, diffusion distance and dissimilarity index to detect community structure. Tested on two artificial networks, the three methods display a high performance.

[3]http://www.cs.cornell.edu/people/tj/svm_light/svm_multiclass.html

2) LDA: LDA is introduced by Blei [3], and it is conceived to analyze the latent thematic structures in the data with large scales, including large collections of text or web documents. We use GibbsLDA++, which is a C/C++ implementation of LDA by using the sampling technique of Gibbs to estimate the parameters and the inference.

3) SMSC: SMSC (Scalable Multi-stage Clustering) is one of the most recent works dealing with the categorization of the tweets [27]. Given a set of tweets S, a set of tags T appearing in S, and $D \subset S$ (such that each $d \in D$ contains at least one tag $t \in T$). The SMSC algorithm starts by (1) creating a whole of virtual documents D', where, each $d^t \in D'$ is a concatenation of all micro-messages in D that contain a specific tag t. The number of messages in D' is equal to the number of hashtags in D. (2) In the second step, it classifies the messages of D' by applying K-Means. (3) Thereafter, each virtual document in D' is retransformed into its original version by assigning each message containing a hashtag to the cluster of the virtual document with which it is associated. Finally, the messages without hashtags are assigned to the closest clusters.

B. Datasets

Two reference corpora are used in the experiments to evaluate the performances of our algorithms.

1) Sander: created by Niek Sanders [25], it is composed of 5513 tweets classified by the author into positive, negative, neutral and without importance. In addition, the corpus is labeled by the following topics: Apple, Google, Microsoft and Twitter. Because of the restrictions of use in Twitter, the contents of the tweets cannot be distributed with the corpus. Thus, the author provides a Python program to download the tweets according to the rules of Twitter. The corpus is on the Sananalytics site[4].

2) TREC 2011 Microblog Track: in TREC 2011, a new task called Microblog Track is introduced to provide a benchmark for research in twitter [17]. This collection contains a sample of tweets over a period of about two weeks spanning from January 24th, 2011 to February 8th, 2011. The TREC 2011 Microblog Track collection is used to evaluate the participating real-time Twitter search systems over 50 official topics. For the evaluation, we extract the tweets written in English posted on January 24th. Then, by using $hashtags$, we extract the tweets concerning seven different topics.

V. RESULTS

In this section, we report the obtained results. We use four evaluation methods namely: Recall, Precision, F-Measure [23] and Rand-Index (RI) [22].

[4]http://www.sananalytics.com/lab/twitter-sentiment/

A. Results of GLIC

1) Results of Grouping Like-Minded Users: As we said in the section III-A1, we test two different methods of data normalization. Let "GLIC-Norm 1" (respectively "GLIC-Norm 2") designates the GLIC algorithm when applying the equation (1) (respectively equation (2)), and "GLIC-WN" designates our algorithm without normalizing data. Tables I and II show the achieved clustering accuracy.

The textual data in social networks are characterized by sparseness. This sparseness is due to the variety of lexical fields used in social networks and to the limitation of publication length (140 characters in Twitter). Thus, the occurrence matrix built in III-A1 is dominated by zeros. This characteristic presents one of the major challenges in social network analysis.

Table I
RESULTS OF GROUPING LIKE-MINDED USERS IN THE SANDER CORPUS USING THE FIRST METHOD

	Recall	Precision	F-measure	RI
K-Means	0.51	0.37	0.42	0.45
LDA	0.51	0.51	0.51	0.68
SMSC	0.85	0.75	0.80	0.76
GLIC-WN	0.57	0.46	0.5	0.56
GLIC-Norm 1	0.67	0.45	0.54	0.56
GLIC-Norm 2	0.91	0.9	0.91	0.9

Table II
RESULTS OF GROUPING LIKE-MINDED USERS IN THE TREC 2011 CORPUS USING THE FIRST METHOD

	Recall	Precision	F-measure	RI
K-Means	0.81	0.48	0.6	0.68
LDA	0.52	0.57	0.54	0.79
SMSC	0.86	0.83	0.85	0.91
GLIC-WN	0.92	0.92	0.92	0.92
GLIC-Norm 1	0.9	0.85	0.87	0.9
GLIC-Norm 2	0.94	0.94	0.94	0.94

According to the obtained results, we remark that K-Means and LDA deliver low performances on both corpuses, compared to SMSC and GLIC-Norm 2. Those performances highlight the limits of K-Means and LDA toward the data sparseness.

The SMSC algorithm displays a high performance (F-measure equal to 0.8 in the Sander corpus and 0.85 in TREC 2011 corpus), which prove the effectiveness of this method. But the higher performances are obtained with GLIC-Norm 2 (over 0.9 for all metrics in both corpuses). The high obtained accuracies are due to the use of PCA to extract the interest centers. In fact, PCA allow extracting the axes where data are concentrated. Projecting the original data in the interest centers space reduces the data noise and the sparseness effect.

Considering the variance between the results of GLIC-WN and GLIC-Norm 1 in both corpuses, we notice that the GLIC algorithm is not stable without normalization. However, our algorithm is stable and gives a better

Table III
RESULTS OF CLASSIFICATION USING THE SVM CLASSIFIER

	Recall	Precision	F-measure
Sander	0.93	0.93	0.93
TREC 2011	0.95	0.93	0.94

performances with normalization, especially using the equation (2).

2) Users Distribution by Clusters: Given that our goal is to group users sharing the same interests, the optimal result is homogeneous groups where each one contains tweets evoking one single interest center. In this part, we evaluate the homogeneity of the obtained groups. Figure 2 shows the users distributions. The sub-figures (a), (b), (c) and (d) are the users distributions obtained respectively by K-Means, LDA, SMSC and GLIC.
We assign a color to each subject in the Sander's corpus. The optimal distribution is the one which is single color, which implies one interest center by group.

In the sub-figure (a) corresponding to the users distribution obtained by K-Means, almost all users are grouped in the fourth cluster, while those users have different interest centers, and cannot be grouped as like-minded. Considering the sub-figure (b), the clusters given by LDA are heterogeneous. Each cluster contains users talking about four different subjects. Also, we cannot affirm the dominating subject of each cluster. With the SMSC algorithm (sub-figure (c)), we obtain three homogeneous clusters (clusters 1, 2 and 4). But the cluster 3 contains considerable proportions of the four subjects of the Sander corpus. Only GLIC (sub-figure (d)) provides four homogenous clusters which prove its effectiveness compared to the baseline algorithms.

3) Results of Classification of New Users: In order to evaluate the performances of the SVM classifier, we divide each corpus into two parts. The first part presents 70% of the entire corpus. We use this part to learn the SVM classifier. The second part (30% of the entire corpus) is used for the test. Table III shows the performances of the SVM classifier. We notice the high values of recall, precision and F-measure. All values exceed 0.93. Those high results prove that our algorithm succeed to affect users to the right classes.

B. Results of GLUCA

In this paragraph we evaluate the GLUCA method performances. We divide the results analysis into quantitative analysis and qualitative analysis.

1) Quantitative Analysis: Tables IV and V show the user grouping results obtained with the GLUCA method. We note that the majority of the values of recall, precision and F-measure exceeds 0.9. These values prove that GLUCA is able to assign the majority of users to the right group. High RI values show that our approach is able to group users sharing the same interests into communities. We can

Table IV
RESULTS OF GROUPING LIKE-MINDED USERS IN THE SANDER CORPUS USING THE SECOND METHOD

	Recall	Precision	F-measure	RI
Without normalization	0.92	0.92	0.92	0.93
Normalization 1	0.92	0.92	0.92	0.93
Normalization 2	0.91	0.9	0.9	0.9

Table V
RESULTS OF GROUPING LIKE-MINDED USERS IN THE TREC 2011 CORPUS USING THE SECOND METHOD

	Recall	Precision	F-measure	RI
Without normalization	0.99	0.99	0.99	1
Normalization 1	0.91	0.90	0.91	0.96
Normalization 2	0.92	0.80	0.86	0.95

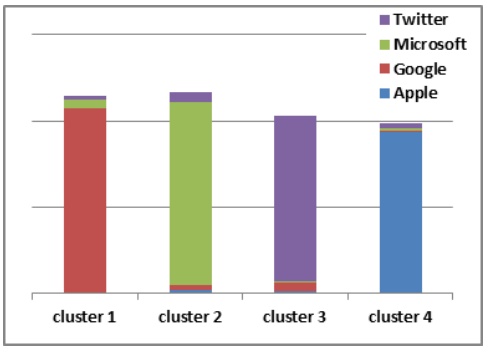

Figure 3. Sanders' Tweets distribution by topics and clusters obtained with the second proposed algorithm

also notice that the obtained results with GLUCA are better than those obtained with the baseline algorithm, and GLIC method. This improvement compared to GLIC can be explained by the fact that we retrieve directely the axes where the correlation between users is maximal, which implies retrieving groups of like-minded users. Also, the K-Means algorithm used for clustering in the GLIC method influence the classification results.

2) Qualitative Analysis: To assess the quality of users found by the second method groups, we draw a graph showing the distribution of users by groups. Figure 3 shows the distribution of users in the corpus Sander obtained with the second method. We note a large homogeneity (example: the majority of users classified at the $cluster$ 1 are talking about Google).

VI. CONCLUSION

In this paper, we presented two algorithms for grouping like-minded people. The main idea of the proposed algorithms is to use PCA to retrieve user communities based on their textual posts. The first algorithm GLIC consists in retrieving latent interest centers from users' posts, and group users according to the retrieved centers. After grouping like-minded users, we use an SVM classifier to classify new users into the created communities. The principle of the second

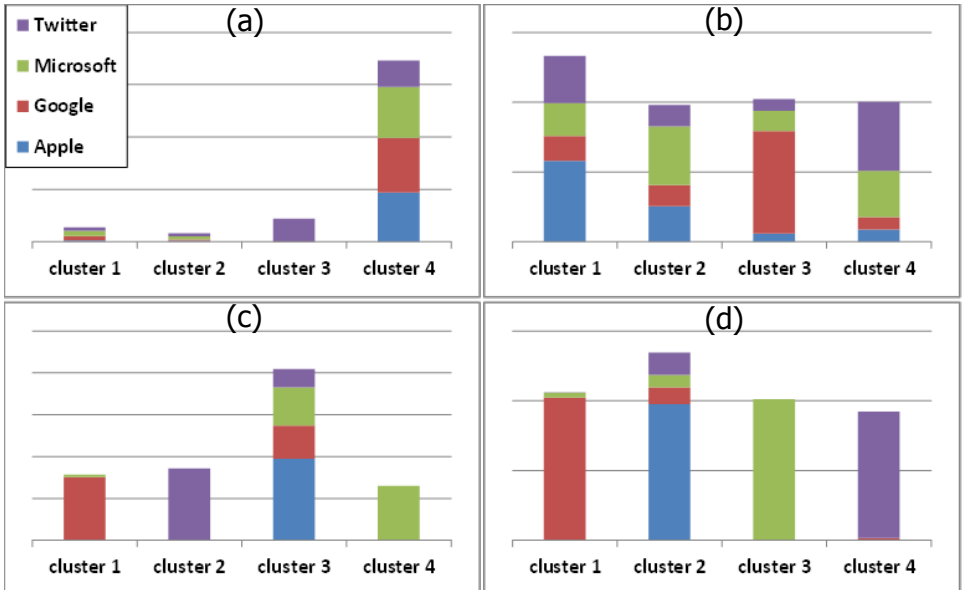

Figure 2. Sanders' Tweets distributions by topics and clusters obtained with the first proposed algorithm and baseline methods

method GLUCA is to retrieve the user concentration axes instead of words concentration axes used in GLIC. Then, we group users which are closer to the same axis into the same group. Finally, we classify new users into communities using an SVM classifier.

To evaluate the proposed algorithms performances, we use two tweets corpus, and we compare the clustering results with three baseline methods, namely: K-Means, LDA and SMSC. The obtained results prove the effectiveness and the high quality of users' communities generated by our algorithms. Also, the proposed algorithms succeed to classify new users into the appropriate groups.

As we start by clustering tweets instead of users, and then we replace each tweet by its editor, one user can be assigned to more than one group. And so, we take into account the case of overlaps between communities.

Amongst the prospects which can be considered is to add link information between users. We can also enhance a semantic layer by integrating an ontology or Folksonomies. Finally, we plan to group like-minded users using different languages.

REFERENCES

[1] Lylia Abrouk, David Gross-Amblard, and Damien Leprovost. Découverte de communautés par analyse des usages. In *EGC 2010 workshops (workshop web social)*, pages A5–5–A5–16, 2010.

[2] Lada A. Adamic and Natalie Glance. The political blogosphere and the 2004 u.s. election: Divided they blog. In *Proceedings of the 3rd International Workshop on Link Discovery*, LinkKDD '05, pages 36–43, New York, NY, USA, 2005. ACM.

[3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[4] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.

[5] Jonathan Gemmell, Andriy Shepitsen, Bamshad Mobasher, and Robin Burke. Personalizing navigation in folksonomies using hierarchical tag clustering. In Il-Yeol Song, Johann Eder, and Tho M. Nguyen, editors, *Data Warehousing and Knowledge Discovery*, volume 5182 of *Lecture Notes in Computer Science*, chapter 19, pages 196–205. Springer, Berlin, Heidelberg, 2008.

[6] Lilia Hannachi, Ounas Asfari, Nadjia Benblidia, Fadila Bentayeb, Nadia Kabachi, and Omar Boussaid. Community extraction based on topic-driven-model for clustering users tweets. In Shuigeng Zhou, Songmao Zhang, and George Karypis, editors, *Advanced Data Mining and Applications*, volume 7713 of *Lecture Notes in Computer Science*, pages 39–51. Springer Berlin Heidelberg, 2012.

[7] Soufiene Jaffali and Salma Jamoussi. Principal component analysis neural network for textual document categorization and dimension reduction. In *Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on*, pages 835–839, 2012.

[8] Soufiene Jaffali, Salma Jamoussi, and Abdelmajid Ben Hamadou. Grouping like-minded users based on text and sentiment analysis. In *Computational Collective Intelligence. Technologies and Applications - 6th International Conference, ICCCI 2014, Seoul, Korea, September 24-26, 2014. Proceedings*, pages 83–93, 2014.

[9] Xin Li, Lei Guo, and Yihong Eric Zhao. Tag-based social interest discovery. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 675–684, New York, NY, USA, 2008. ACM.

[10] Yu-Ru Lin, Hari Sundaram, Yun Chi, Jun Tatemura, and Belle Tseng. Discovery of blog communities based on mutual awareness. In *in: Proceedings of the 3rd Annual Workshop on the Weblogging Ecosystem*, 2006.

[11] Jian Liu. Comparative analysis for k-means algorithms in network community detection. In Zhihua Cai, Chengyu Hu, Zhuo Kang, and Yong Liu, editors, *ISICA (1)*, volume 6382 of *Lecture Notes in Computer Science*, pages 158–169. Springer, 2010.

[12] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[13] Andrew McCallum, Xuerui Wang, and Andrés Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *J. Artif. Int. Res.*, 30(1):249–272, October 2007.

[14] Mary McGlohon, Leman Akoglu, and Christos Faloutsos. Statistical properties of social networks. In *Social Network Data Analytics*, pages 17–42. 2011.

[15] Matthew Michelson and Sofus A. Macskassy. Discovering users' topics of interest on twitter: A first look. In *Proceedings of the Fourth Workshop on Analytics for Noisy Unstructured Text Data*, AND '10, pages 73–80, New York, NY, USA, 2010. ACM.

[16] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review*, E 69(026113), 2004.

[17] Iadh Ounis, Jimmy Lin, and Ian Soboroff. Overview of the trec2011 microblog track. In *TREC*, 2011.

[18] Diana Palsetia, Md. Mostofa Ali Patwary, Kunpeng Zhang, Kathy Lee, Christopher Moran, Yves Xie, Daniel Honbo, Ankit Agrawal, Wei-keng Liao, and Alok Choudhary. User-interest based community extraction in social networks. In *The 6th SNA-KDD Workshop 12*. ACM, 2012.

[19] Symeon Papadopoulos, Yiannis Kompatsiaris, Athena Vakali, and Ploutarchos Spyridonos. Community detection in social media. *Data Min. Knowl. Discov.*, 24(3):515–554, May 2012.

[20] K. Pearson. On lines and planes of closest fit to points in space. *Philosophical Magazine*, 2, 1901.

[21] Marco Pennacchiotti and Siva Gurumurthy. Investigating topic models for social media user recommendation. In *Proceedings of the 20th International Conference Companion on World Wide Web*, WWW '11, pages 101–102, New York, NY, USA, 2011. ACM.

[22] W.M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[23] C. J. Van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979.

[24] Mrinmaya Sachan, Danish Contractor, Tanveer A. Faruquie, and L. Venkata Subramaniam. Using content and interactions for discovering communities in social networks. In *Proceedings of the 21st international conference on World Wide Web*, WWW '12, pages 331–340, New York, NY, USA, 2012. ACM.

[25] Niek J. Sanders. *Sanders-Twitter Sentiment Corpus*. Sanders Analytics LLC, October 2011.

[26] Durgesh K. Srivastava and Lekha Bhambhu. Data classification using support vector machine. *Journal of Theoretical and Applied Information Technology*, 12(1), February 2010.

[27] Oren Tsur, Adi Littman, and Ari Rappoport. Efficient clustering of short messages into general domains. In Emre Kiciman, Nicole B. Ellison, Bernie Hogan, Paul Resnick, and Ian Soboroff, editors, *ICWSM*. The AAAI Press, 2013.

[28] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[29] Xufei Wang, Huan Liu, and Wei Fan. Connecting users with similar interests via tag network inference. In *the 20th ACM Conference on Information and Knowledge Management (CIKM)*, Glasgow, Scotland, UK, 2011.