Multi-view Clustering in a Social Network

Gustavo Guedes[1,2], **Eduardo Bezerra**[1,2], **Geraldo Xexéo**[2,3]

[1]CEFET/RJ - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca
Av. Maracanã, 229 - Rio de Janeiro - RJ - Brazil

[2]COPPE/UFRJ - Programa de Engenharia de Sistemas e Computação - UFRJ
Caixa postal 68511, CEP 21945-970 - Rio de Janeiro - RJ - Brazil.

[3]IM/UFRJ - Instituto de Matemática - UFRJ - Rio de Janeiro - RJ - Brazil.

gustavo.guedes@cefet-rj.br, ebezerra@cefet-rj.br, xexeo@cos.ufrj.br

Abstract. *An important problem in social network analysis is the partitioning of its users, in order to discover groups of user that have common interests or characteristics. Given a collection of objects to be clustered, typically there is not a single way of forming the partitions. Besides, when objects are users of a social network, each object may be present in several datasets associated to the social network. This paper describes work in progress on a procedure to cluster users of a social network. The particular characteristics of the social network we study motivates us to use a relatively new approach to clustering, namely multi-view clustering. We present our multi-view clustering procedure, that uses several existing perspectives of these users to generate alternative and complimentary clusterings, that in turn can reveal novel ways of interpreting these users.*

1. Introduction

Online social networks are ubiquitous in our modern society. Through them, a particular user may, for example, share his/her opinions and states of mind with his/her associated users. Users in a social network can also identify with each other, which results in several structures of association. Given a social network, an interesting problem is to cluster its users, in order to discover patterns associated to their common behaviors, characteristics, interests, etc [Wasserman and Faust 1994]. On the other hand, the clustering community has recently acknowledged that, given a collection of objects to cluster, typically there is not a single way to partition them [Muller et al. 2012]. Indeed, a collection of objects can be clustered in several alternative and complimentary ways, and each one of them may reveal a different and interesting perspective. This is particularly relevant in social networks, in which each user may be part of several datasets with representations of varied complexity. For example, each user may be either represented by the set of messages he/she posts, or by the data he/she provides in a personality test, or even by his/her corresponding patterns of interaction with other users.

Several clustering algorithms have recently been proposed to form multiple partitions of a collection of objects. For a review, we direct the reader to [Muller et al. 2012] and references therein. However, to the best of our knowledge, there is no attempt to apply the multi-clustering approach to data associated to an online social network. This motivates us to investigate the application of a multi-clustering procedure to cluster users

of an existing social network. In particular, we use *Meu Querido Diário* (MQD for short) a Brazilian social network in which users can register and talk about their day-to-day activities. Hence in this work we aim at presenting our multi-view clustering procedure, that uses several existing perspectives of data associated to users of MQD to generate alternative, non-redundant clusterings, that in turn can reveal novel ways of viewing and interpreting these users in a data analysis setting.

The remainder of this paper is organized as follows. In Section 2, we present a brief introduction to clustering, with focus on the multiple-clustering task. In Section 3, we present details about the MQD social network. A description of our multi-clustering method to partition the collection of MQD users in several non-redundant ways is presented in Section 4. Finally, in Section 5 we conclude.

2. Clustering

Clustering is a popular data mining task often used as a first step in the analysis of complex datasets. Most clustering algorithms can be thought as a discrete optimization process driven by an objective-function. Typically, the corresponding search space is huge where the states are all the possible partitionings of the input objects. The optimization process tries to find a partitioning of the input objects in such a way that similar objects are put in the same cluster, while dissimilar objects are located in different clusters. The final result is a partitioning of the data that exposes of particular view of the data.

Most clustering algorithms find one particular partitioning of the available data [Jain et al. 1999]. However, multi-faceted data is relatively common nowadays, which gives rise to the possibility of generating several non-redundant partitionings of the same data. Hence, several multi-clustering algorithms have been proposed recently.

The main goal of a multi-clustering algorithm is to generate several clustering solutions from the same collection of objects. Although the literature on single-solution clustering algorithms is overwhelming, only recently the problem of generating multiple partitionings has received more attention.

According to [Vinh and Epps 2010], existing approaches for multiple clustering can be divided into two broad types: objective-function-oriented and data transformation-oriented. In the first one, the clustering is guided by a diversity-aware objective function, which drives the search process away from one or multiple existing target clusterings. In the data transformation-oriented approach, the clustering process is mainly guided by a data transformation prior to using a regular clustering algorithm. This data transformation aims at revealing a novel perspective of the data that was hidden in the original representation of the data. Our present work fits in the former approach, as we explain in Section 4.

3. The MQD Social Network

MQD is an online social network for Brazilian users. This social network is available at `http://www.meuqueridodiario.com.br/`, and allows each user to post entries and associate *emotions tags* to them. When using MQD, users often describe what they did during the day, what they are feeling or some description about their emotional state. They can choose one of six emotions to associate to their entries (posts).

Each posted message may be tagged with one of six basic emotions proposed by Ekman [Ekman and Friesen 1978]: *anger*, *disgust*, *fear*, *sadness*, *happiness* and *surprise*.

Furthermore, users of the MQD social network can answer a *personality test* [Andrade 2008], which is based on a Brazilian representation of the Five Factor Model of personality traits proposed by [Piedmont 1998]. This test measures five traits in personality: *openness*, *conscientiousness*, *extraversion*, *agreeableness* and *neuroticism*. The test is composed by an inventory of 44 questions related to the big five personalities. The user chooses a number from 1 to 5, where 1 denotes strong disagreement, numbers 2, 3 and 4 represent intermediate judgments and 5 denotes strong agreement. Each personality factor for a given user is calculated as following:

- Extraversion: 1, 6R, 11, 16, 21R, 26, 31R, 36.
- Agreeableness: 2R, 7, 12R, 17, 22, 27R, 32, 37R, 42.
- Conscientiousness: 3, 8R, 13, 18R, 23R, 28, 33, 38, 43R.
- Neuroticism: 4, 9R, 14, 19, 24R, 29, 34R, 39.
- Openness: 5, 10, 15, 20, 25, 30, 35R, 40, 41R, 44.

In the above list, 'R' denotes reverse-scored items. In order to illustrate the computation of factors, consider that an user provided responses as shown in Table 1 when answering the personality test.

Table 1. Responses for some questions of personality test

Question	Score
1	3
6	4
11	2
16	3
21	1
26	5
31	4
36	3

We can see that this user has scored 3 for question 1, 4 for the question 6, and so on. We can calculate the extraversion score as in Eq. 1. Accordingly, the user has the value 3.125 representing her/his extraversion factor. The other personality factors are calculated in an analogous way.

$$\frac{(3 + (6 - 4) + 2 + 3 + (6 - 1) + 5 + (6 - 4) + 3)}{8} = 3.125 \tag{1}$$

Additionally, the user information collected in MQD comprises age, sex, birth date, state, marital status, etc. When the user writes a new post, she/he needs to fill the title, text, event date and emotion tag. The emotion tag is not required. Besides, the system records some other information like written date, comment number of the post and obviously the user who wrote that post. As is common in online social networks, users may write comments in each other posts. Each post in MQD may have several comments. Currently, more than $7,000$ MQD users have answered the personality test. Aditionally,

MQD has more than $19,000$ friend relationships and approximatelly $20,000$ posts with associated emotion tags. The number of comments is curretly $64,450$.

4. Multi-clustering method

We are now ready to describe our multi-view clustering procedure applied to MQD social network. In our clustering model, objects to be clustered are users of this social network. Formally, each user is represented as a triple $(\vec{e}, \vec{p}, \vec{w})$ of vectors in vector spaces E, P and W, about emotion tags, personality traits and message posts, respectively. Herein, we describe these vector spaces and how a user is represented in each one of them.

First of all, recall that, when posting a message in MQD, each user tags it with an associated emotion. We associate a number j, $1 \leq j \leq 6$, to each possible emotion tag. Therefore, the i-th user is represented as a vector $\vec{e}_i = (e_{i1}, e_{i2}, \ldots, e_{im})$, where e_{ik} is the number of messages tagged with emotion k by user i. Analogously, we represent each user in vector space P as a vector $\vec{p}_i = (p_{i1}, p_{i2}, \ldots, p_{in})$. Each dimension in vector $\vec{p}_i$ corresponds to the factor assigned to user i for a given personality trait.

Finally, in order to represent the i-th user in W, we use a vector space model [Manning et al. 2008]. For this, we first compute the collection of words used in at least one message posted by the set of users in MQD. As a preprocessing step, we remove *funcion words* (e.g., preprosition, articles, etc.) from this set. We also apply stemming to reduce words to their root form. Let T be the resulting collection of words after preprocessing, and let $|T| = q$. Then, for each user, we compute his/her corresponding vector in W-space by using a well-known weighting measure in Information Retrieval, namely, TF-IDF [Manning et al. 2008]. Hence, the i-th user is represented in W-space as a vector $\vec{w}_i = (w_{i1}, w_{i2}, \ldots, w_{iq})$, in which component w_{ij}, $1 \leq j \leq q$, is computed by using TF-IDF measure. Given the user u_i and the word t_j, we compute w_{ij} by using Eq. 2.

$$w_{ij} = tf(u_j, t_j) \times idf(t_j) = tf(u_i, t_j) \times \log\left(\frac{|\mathcal{U}|}{df(t_j)}\right) \tag{2}$$

In Eq. 2, $|\mathcal{U}|$ corresponds to the total number of users, $tf(u_i, t_j)$ is the number of times that term t_j occurs in the set of messages posted by user u_i, and $df(t_i)$ is the number of users that used word t_i at least once in her/his posted messages. This way, w_{ij} is a number that reflects how important is word t_j in the overall content posted by user u_i in the social network.

The above described three-part user representation is the input for our multi-view clustering procedure, as we explain herein. Our goal is to generate alternative and non-redundant clusterings from the collection of MQD users. As the basic clustering algorithm, we use the well studied k-means algorithm [MacQueen 1967]. Given a collection of users $\mathcal{U}$ and a number k as input, k-means generates k-partition of $\mathcal{U}$, that is, a clustering comprised of k non-overlaping clusters. In k-means, the main goal is to find k centroids (one for each cluster) by locally optimizing an objective function.

Our main idea is to take one of the three user views (E, P, W), to generate a first clustering of users. This first clustering is then used to guide the generation of other clusterings from the remaining views, as we explain in the following. Let us call this first clustering the *base clustering*. We use classical k-means to generate the base clustering.

After that, we generate two other clusterings, taking each one of the two other views as input. Now, to generate these two other clusterings, we modify k-means objective function in order to penalize solutions that are similar to the base clustering solution. The aim of this modification in the objective function is to take into account the constraint that the generated clustering solution must be *dissimilar* to the base clustering.

Let $\mathcal{U} = \{u_i\}$ be the set of users. Also, consider that each cluster in the base clustering has an assigned label $l \in L$, where $L = \{1, 2, \ldots, k\}$, that $c(u) : \mathcal{U} \rightarrow L$ is a function that returns the label of the cluster associated to a given user, and that μ_l is the centroid of cluster C_l. The objective function we use to generate the two other clusterings is defined by Eqn. 3.

$$\begin{aligned} \text{Obj} \;\; = \;\; & \sum_{u_i \in \mathcal{U}} \text{sim}(u_i, \mu_{c(u_i)}) \\ & - \; \text{TotalCostML} \\ & - \; \text{TotalCostCL} \end{aligned} \tag{3}$$

In the first part of this objective function, $\text{sim}(u_i, \mu_{c(u_i)})$ is a function that measures the similarity between user u_i and the corresponding centroid $c(u_i)$. The choice of the similarity measure to use depends on the view selected to generate the base clustering. Two possible choices are the Euclidean distance and the cosine distance. In particular, if the E view is used to generate one of the two remaining clusterings, we use cosine distance (instead of the euclidean distance) due to the fact that the former is more appropriate to sparse data in high dimensions such as post messages represented in a vector space as we find in W-space.

The second and third parts of this objective function correspond to the costs of violating constraints sampled from the base clustering. To generate a constraint in this set, we randomly select a pair (u_i, u_j) of users and query their associated labels in the base clustering. If they have equal labels, we define a *cannot-link constraint*, that is, a constraint indicating that u_i and u_j must be put in different cluster when generating the remaining clusterings. Analogously, if the pair (u_i, u_j) has different labels, we define a *must-link constraint*, that is, a constraint indicating that u_i and u_j must be put in the same cluster when generating the remaining clusterings. Eqns. 4 and 5 give the total costs of violating must-link and cannot-link constraints, respectively.

$$\text{TotalCostML} = \sum_{(u_i, u_j) \in \mathcal{S}_{ML}} v_{ij}^{ML} \times I(c(u_i), c(u_j)) \tag{4}$$

$$\text{TotalCostCL} = \sum_{(u_i, u_j) \in \mathcal{S}_{CL}} v_{ij}^{CL} \times I(c(u_i), c(u_j)) \tag{5}$$

In Eqns. 4 and 5, $I(x, y)$ is the indicator function (i.e., I returns 1 when $x = y$, and I returns 0 when $x \neq y$). The values of v_{ij}^{ML} and v_{ij}^{CL} are the costs of violating must-link and cannot-link constraints, respectively. Notice that the role TotalCostCL and TotalCostML in the objective function is to penalize clustering solutions that are

similar to the base solution. The amount of constraints extracted from the base clustering is a parameter of our multi-view clustering method.

5. Conclusions

Our present work is preliminary, since here we considered the structure and metadata of single social network (MQD). However, we believe our method can be generalized to other social networks since most of them present multi-faceted data. We are currently setting up our experiment environment in order to validate our method. We believe these experiments will provide further information that we can use do adequately determine the amount of constraints necessary to generate the remaining clusterings from the base clustering.

References

Andrade, J. M. (2008). *Evidências de Validade do Inventário dos Cinco Grandes Fatores de Personalidade para o Brasil.* doctor thesis, Universidade de Brasília.

Ekman, P. and Friesen, W. (1978). *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Consulting Psychologists Press, Palo Alto.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323.

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

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

Muller, E., Gunnemann, S., Farber, I., and Seidl, T. (2012). Discovering multiple clustering solutions: Grouping objects in different views of the data. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 1207–1210, Washington, DC, USA. IEEE Computer Society.

Piedmont, R. (1998). *The Revised NEO Personality Inventory: Clinical and Research Applications*. The Springer Series in Social Clinical Psychology. Springer.

Vinh, N. X. and Epps, J. (2010). mincentropy: A novel information theoretic approach for the generation of alternative clusterings. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 521–530.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Structural Analysis in the Social Sciences. Cambridge University Press.