2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Opinion retrieval through unsupervised topological learning

Nicoleta Rogovschi Nistor Grozavu

***Abstract*— Opinon Mining is the field of computational study of peopel's emotional behavior expressed in text. The purpose of this article is to introduce a new framework for emotion (opinion) mining based on topological unsupervised learning and hierarchical clustering.**

In contrast to supervised learning, the problem of clustering characterization in the context of opinion mining based on unsupervised learning is challenging, because label information is not available or not used to guide the learning algorithm. The algorithm described in this paper provides topological clustering of the opionon issued from the tweets, each cluster being associated to a prototype and a weight vector, reflecting the relevance of the data belonging to each clsuter. The proposed framework requires simple computational techniques and are based on the double local weighting self-organizing map (dlw-SOM) model and Hierarchical Clustering.

The proposed framework has been used on a real dataset issued from the tweets collected during the 2012 French election compaign.

I. Introduction

Opinion Mining is a recent research field in science that combines informational retrieval and computational linguistics. This field is an emerging problem in data mining and only some work on this subject can be found in the literature, especially using unsupervised machine learning techniques.

In recent years, there is a growing interest in sharing personal opinions on the Web, such as product reviews, photos, videos, economic analysis, political polls, etc. These informations can be found in discussion forums, tweets, social networks, etc. These opinions cannot only help independent users make decisions, but also obtain valuable feedbacks [1]. The opinion mining research field, including sentiment classification, opinion extraction, opinion question answering, and opinion summarization, etc. are receiving growing attention [2].

Opinion retrieval from text data is very diferent to classical informational retrieval approaches. Firstly, relevant documents should not only be relevant to the targets, but also contain subjective opinions about them. Secondly, the text collections are more informal word of mouth web data. Typical sources are blogs that generally reflect personal opinions, forums that present group opinions and tweets data where the messages are more shortly represented and the analysis became more difficult. Thirdly, although web retrieval pays more attention to precision, opinion retrieval attaches extra importance to recall, since further sentiment mining relies heavily on the coverage of the opinion collection [3]. Finally, the greatest challenge for opinion retrieval approaches lies in the difficulty in representing the users information need and to characterize the opinions group in an automatic way by detecting the relevant features.

Nicoleta Rogovschi is within LIPADE University of Paris 5, France nicoleta.rogovschi@parisdescartes.fr

Nistor Grozavu is within LIPN-UMR CNRS 7030, Université de Paris 13, Sorbonne Paris Cité, FRANCE, nistor.grozavu@@lipn.univ-paris13.fr

Opinions are central to almost all human activities because they are key influencers of our behaviors. Whenever we need to make a decision, we want to know others opinions [4]. In the real world, businesses and organizations always want to know more about the public opinions about their products and services in order to better organize their offers. The opinions are also important for the individual consumers what want to know the opinions of other users about a product before purchasing it, or about a discussion before to make a conclusion. In a political election, the individuals can be also interested in the others opinions about political candidates before making a voting decision [4], [5].

In this paper, we focus on opinion retrieval, whose goal is to find a set of tweets containing not only the similar query keyword(s) but also the relevant emotions and to make an automatical characterization of the opinions' groups (clusters). One of the chellenge in this case is the representation of information needs for effective opinion retrieval.

In recent years, we have witnessed that opinionated postings in social media have helped reshape businesses, and sway public sentiments and emotions, which have profoundly impacted on the social and political systems. Such postings have also mobilized masses for political changes such as those happened in some Arab countries in 2011. It has thus become a necessity to collect and study opinions on the Web [4].

In this work, we are interested in methods which aims at automatically finding attitudes or opinions about specific targets, in our case the opinios about the candidates in 2012 french elections.

The rest of the paper is organized as follows: the proposed framework for the opinion mining is presented in Section 2. We introduce the weighted topological learning in section 2.B after the Preprocessing step presented in section 2.A. In Sections 3, we present the validation of the proposed approach on a tweets data sets and finally the paper ends with a conclusion and some future works for the proposed framework.

II. Opinion Mining

Data Clustering is the main task of knowledge discovery in databases. It aims to group a set of objects in such a way that objects in the same group (called cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters).

The approaches allowing the extraction of opinions from the text can be categorized into two main groups: lexicon-based and classification-based.

The lexicon based approaches uses a manually or automatically built list of subjective words, such as ‘good’ and ‘like’, and assumes that the presence of these words in a document (tweet) is the evidence of document opinionatedness. A term’s opinion score can be used in diferent ways to assign an opinion score to the whole document. The classification-based approaches implies the use of the word occurrence and sometimes linguistic features and builds a classifier based on positive (opinionated) and negative (non-opinionated) documents using Machine Learning techniques. Nevertheless, most of the early research in this area ignore the problem of retrieving documents that are related to the topic of the user’s interest [6].

For this work, we propose to use the linguistic knowledge and the topological clustering in order to obtain clusters of opinions and to automaticaly characterize the opinions. The figure 1 shows the proposed framework.

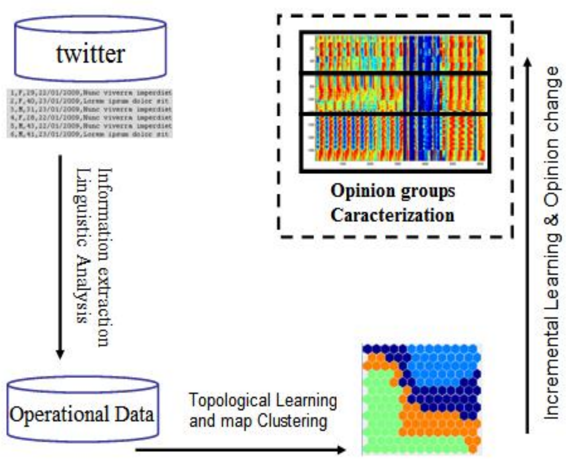

Fig. 1. The proposed framework for opinion mining

Also, the proposed framework can be used in incremental way as the topological map can be updated using new data (tweets) after the learning process.

In the next sections we describe the three steps used in the proposed framework: prepocessing, topological learning and opinion clustering and characterization. Note, that all these steps are linked and can not be used sepparetly for this problem.

A. Preprocessing

For the preprocessing step, we, firstly start by annotating the tweets using a morphosyntactic tag that allows to assign to each term of a tweet a part of speech (POS) tag. Then, the principle of Bag of Words is used in order to create a bag of words from the tweets by extracting the words (terms) from each tweet (document).

And, the last part of the preprocessing step is the use of the TF-IDF [7].

The TF-IDF weight (term frequency-inverse document frequency) is a weight often used in text mining. This weight criterion is a statistical measure used to evaluate the importance of a term from a document in a corpus. The importance increases proportionally to the number of times a term appears in the document but is offset by the frequency of the word in the respective collection [7].

For a term t_i from the document d_j, its term frequency (TF) is defined as follows:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{1}$$

where n_{ij} represents the number of occurrences of the term t_i in the document d_j.

The Inverse Document Frequency (IDF) is a measure that compute the importance of the term t_i in the respective collection (corpus) which is obtained by computing the logarithm of the inverse of the proportion of documents in the corresponding collection. The IDF is defined as follows:

$$idf_i = log_2 \frac{|D|}{|d_j : t_i \in d_j|} \tag{2}$$

where $|D|$ is the total number of documents presented in the corpus, and $d_j : t_i \in d_j$ represents the documents containing the term t_i.

And, finally, the TF-IDF weight of a term t_i is the product of TF and IDF:

$$TF - IDF_{i,j} = TF_{i,j} \times IDF_i \tag{3}$$

B. Topological clustering

Data mining, or knowledge discovery in databases (KDD), an evolving area in information technology, has received much interest in recent studies. The aim of data mining is to extract knowledge from data [8]. The data size can be measured in two dimensions, the size of features and the size of observations. Both dimensions can take very high values, which can cause problems during the exploration and analysis of the dataset. Models and tools are therefore required to process data for an improved understanding. Indeed, datasets with a large dimension (size of features) display small differences between the most similar and the least similar data. In such cases it is thus very difficult for a learning algorithm to detect similarity variables that define the clusters [9].

Topological learning is a recent direction in Machine Learning which aims to develop methods grounded on statistics to recover the topological invariants from the observed data points. Most of the existed topological learning approaches are based on graph theory or graph-based clustering methods. The topological learning is one of the most known technique which allow clustering and visualization simultaneously. At the end of the topographic learning, the ”similar” data will be collect in clusters, which correspond to the sets of similar observations. These clusters can be represented by more concise information than the brutal listing of their patterns, such as their gravity center or different statistical moments. As expected, this information is easier to manipulate than the original data points. The neural networks based techniques are the most adapted to topological learning as these approaches represent already a network (graph). The models that interest us in this paper are those that could make at the same time the dimensionality reduction and clustering, i.e. using Self-Organizing Maps (SOM) [10] for dimensionality reduction and Hierarchical Clustering to cluster the map. SOM models are often used for visualization and unsupervised topological clustering. Its allow projection in small spaces that are generally two dimensional. Some extensions and reformulations of the SOM model have been described in the literature [11], [12], [13].

We find several important research topics in cluster analysis and variable weighting [14], [15], [16], [17], [18]. In [17], the authors propose a probabilistic formalism for variable selection in unsupervised learning using Expectation-Maximization (EM). Grozavu et al. [9] proposed two local weighting unsupervised clustering algorithms (lwo-SOM and lwd-SOM) to categorize the unlabelled data and determine the best feature weights within each cluster. Similar techniques, based on k-means and weighting have been developed by other researchers [15], [19].

Double local weighting SOM : dlw-SOM

One of the significant limitations of the classical SOM algorithms is that they treat all features equally. This is not desirable for many applications of clustering, in which observations are defined by a large number of features. A cluster provided by SOM is often characterized by only a subset of features rather than by the entire features set. The presence of other features may therefore prevent the discovery of the specific cluster structure associated to each cell. The relevance of each observation and feature changes from one cluster to another.

dlw-SOM provides a principal alternative to classical SOM and overcomes some limitations mentioned previously. Indeed, the proposed clustering algorithm and feature weighting aims to select the optimal prototypes, observations and feature weights at the same time [20]. Each prototype $\mathbf{w}_j = (w_j^1, w_j^2, ..., w_j^m)$ corresponding to cell j is allowed to have its own set of local features weights $\pi_j^{(o)} = (\pi_j^{(o)1}, \pi_j^{(o)2}, ..., \pi_j^{(o)m})$ and its own set of local distance weights $\pi_j^{(d)} = (\pi_j^{(d)1}, \pi_j^{(d)2}, ..., \pi_j^{(d)m})$ respectively. We denote the set of weight vectors ($|\Pi| = |W|$) by $\Pi = \{\pi_j, \pi_j \in \Re^m\}_{j=1}^{|\Pi|}$ for both observation and distance weighting.

For the double local weighting process, we introduce the both weights in the SOM objective function, and we obtain:

$$R_{dlw-SOM}(\chi, \mathcal{W}, \Pi^{(d)}, \Pi^{(o)}) = \sum_{i=1}^{|N|} \sum_{j=1}^{|\mathcal{W}|} \mathcal{K}_{j,\chi(\mathbf{x}_i)} (\pi_j^{(d)})^\beta \|\pi_j^{(o)} \mathbf{x}_i - \mathbf{w}_j\|^2$$

where $\Pi^{(d)}$ are the distance weights, $\Pi^{(o)}$ the observations weights and β is the discrimination coefficient.

As we combined two types of the weighting techniques, contrarily to precedent weighting approaches [9], the minimization of the $R_{dlw-SOM}$ objective function is made in four steps:

1) Minimize $R_{dlw}(\chi, \hat{\mathcal{W}}, \hat{\Pi^{(d)}}, \hat{\Pi^{(o)}})$ with respect to χ by fixing $\hat{\mathcal{W}}$, $\hat{\Pi^{(d)}}$ and $\hat{\Pi^{(o)}}$. The expression is defined as follows:

$$\chi(\mathbf{x}_i) = \arg\min_j \left((\pi_j^{(d)})^\beta \|\pi_j^{(o)} \mathbf{x}_i - \mathbf{w}_j\|^2 \right) \qquad (5)$$

2) Minimize $R_{dlw-SOM}(\hat{\chi}, W, \hat{\Pi^d}, \hat{\Pi^o})$ with respect to W by fixing $\hat{\chi}$, $\hat{\Pi^{(d)}}$ and $\hat{\Pi^{(o)}}$. The prototype's vectors are updated using the following expression:

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \epsilon(t)\mathcal{K}_{j,\chi(\mathbf{x}_i)}(\pi_j^{(d)})^\beta \left(\pi_j^{(o)}\mathbf{x}_i - \mathbf{w}_j(t)\right)$$

3) Minimize $R_{dlw-SOM}(\hat{\chi}, \hat{\mathcal{W}}, \hat{\Pi^{(d)}}, \Pi^{(o)})$ with respect to $\Pi^{(o)}$ by fixing $\hat{\chi}$, $\hat{\mathcal{W}}$ and $\hat{\Pi^{(d)}}$. The update of the observation weights vectors $\pi^{(o)}{}_j(t+1)$ are made using the following expression:

$$\pi_j^{(o)}(t+1) = \pi_j^{(o)}(t) + \epsilon(t)\mathcal{K}_{j,\chi(\mathbf{x}_i)}(\pi_j^{(d)}(t))^\beta \mathbf{x}_i \left(\pi_j^{(o)}(t)\mathbf{x}_i - \mathbf{w}_j(t)\right)$$

4) Minimize $R_{dlw-SOM}(\hat{\chi}, \hat{\mathcal{W}}, \Pi^{(d)}, \hat{\Pi^{(o)}})$ with respect to $\Pi^{(d)}$ by fixing $\hat{\chi}$, $\hat{\mathcal{W}}$ and $\hat{\Pi^{(o)}}$. The update of the distance weights vectors $\pi^{(d)}{}_j(t+1)$ are made using the following expression:

$$\pi^{(d)}{}_j(t+1) = \pi_j^{(d)}(t) + \epsilon(t)\mathcal{K}_{j,\chi(\mathbf{x}_i)}\beta(\pi_j^{(d)}(t))^{\beta-1} \left(\pi_j^{(o)}(t)\mathbf{x}_i - \mathbf{w}_j(t)\right)$$

This expression allows us to optimize variable weights in order to obtain the best clustering by minimizing the ratio of the average within-cluster distortion over the average between-cluster distortion.

C. Hierarchical Clustering

Clustering algorithms are generally classified as partitional clustering and hierarchical clustering, based on the properties of the generated clusters ([21]; [22]; [23]; [24]). Partitional clustering divides data samples into a single partition, whereas a hierarchical clustering algorithm groups data with a sequence of nested partitions.

There is two types of the hierarchical clustering methods: agglomerative approach and divide approach. Divide hierarchical clustering method starts from a cluster which contains all the data and divide this cluster until obtaining the desired clusters. Contrarily, agglomerative hierarchical clustering method starts from n clusters (n data) and will merge these clusters until obtaining a cluster containing the whole data.

For this work we used the Hierarchical Clustering algorithm with Wards criterion to avoid merging empty cells. This procedure will allow us to avoid clustering "cleaning" by eliminating the cells/clusters which have no captured samples.

Agglomerative clustering starts with n clusters, each of which includes exactly one data point. A series of merge operations is then followed that eventually forces all objects into the same group.

III. Experimental results

The work presented in this paper were tested on a tweets dataset which were obtained as a part of the PoloP Project5 (Political Opinion Mining) which aims to cope with the analysis of the evolution of French political communities over Twitter during 2012 both in terms of relevant terms, opinions, behaviors. 2012 is particularly important for French political communities dues the two main elections: Presidential and Legislative. The 6th of May was the final Presidential election where F. Hollande has been elected and the legislative elections were finished one month after [25].

The algorithm dlw-SOM allows us to obtain on the one hand, a two-dimensional projection data and on the other hand, a weighting of variables specific to each region of the map. Vesanto and Alhoniemi (2000) [26] have proposed to segment a topological map by combining the k-means algorithm and Davies-Bouldin index which allows to automatically determine the size of the partition after segmentation. Indeed to use the k-means to cluster the map, we applied the HIerarchical Clustering introduced in section 2 which allows us to obtain stable results compared to k-means. We have applied this approach on referents and on the weights.
We obtained a topological map containing 169 cells (figure 3), and applying the Hierachical clustering on the map, we obtained 3 clusters. Note that initially we clusterd the map from 2 to 10 clusters and we computed the Davies-Bouldin index for each one (presented in Table 1) in order to choose the best clustering result. The DB index [27] is an internal index between two clusters and it's computing as follows: A similarity measure R_{ij} between the clusters C_i and C_j is defined based on a measure of dispersion of a cluster C_i, let s_i, and a dissimilarity measure between two clusters d_{ij}. The R_{ij} index is defined to satisfy the following conditions:

- $R_{ij} \geq 0$
- $R_{ij} = R_{ji}$
- if $s_i = 0$ and $s_j = 0$ then $R_{ij} = 0$
- if $s_j \geq s_k$ and $d_{ij} = d_{ik}$ then $R_{ij} \geq R_{ik}$
- if $s_j = s_k$ and $d_{ij} < d_{ik}$ then $R_{ij} \geq R_{ik}$

So, these conditions impose to R_{ij} to be a non-negative and symmetric. To satisfy the above-mentioned conditions, we have: $R_{ij} = \frac{(s_i+s_j)}{d_{ij}}$.

Then, the DB index is defined as:

$$DB_{nc} = \frac{1}{n_c}\sum_{i=1}^{n_c} R_i \qquad (6)$$

$$and \quad R_i = max_{i=1,...,n_c, i\neq j} R_{ij}, \quad i = 1, ..., n_c \qquad (7)$$

The DB_{nc} is the average similarity between each cluster $c_i, i = 1, ..., n_c$ and its most similar one. So, we seek clusterings that minimize the DB, and thus have minimum possible similarity with the clusters. Some variants of this index were proposed in literature which are based on Minimum Spanning Tree (MST), Relative Neighborhood Graph (RNG) and the Gabriel Graph (GC) concepts.

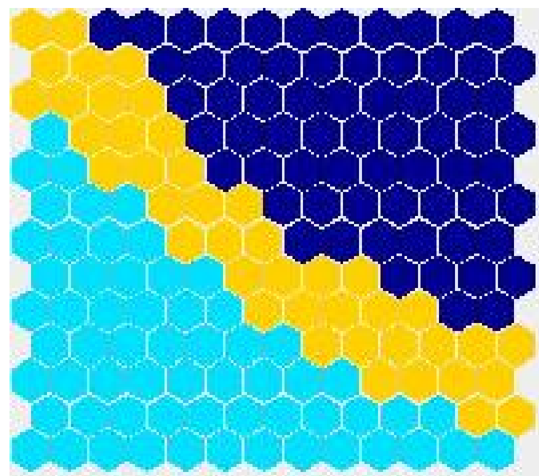

Fig. 2. Map segmentation using HAC on referents vector

TABLE I
Davies-Bouldin index for each clustering result

nb cl.	2	3	4	5	6	7	8	9
DB index	0.56	0.41	0.48	0.53	0.47	0.59	0.62	0.57

In the table 2 we show an example of the pertinant terms from the tweets of the both opinion clusters that characterize them. Note that the cluster situated in the midle of the map (the yellow cluster) contains similar opinions from other two clusters due to the neighborhood of the map, and it seems that tweets belonging to this cluster contains a neutral opinion.

These results (the relevant terms for each opinion cluster translated from French) are relevent with the real opinion of peoples about this election campaign.

IV. Conclusion

In this study we proposed a framework for opinon mining based on topological unsupervised learning and hierarchical clustering. The algorithm described in this paper provides

TABLE II

PERTINENT TERMS FOR THE OPINION CLUSTERS

cluster	1	2
terms	lost the Triple A Holland will love europe growing device change is now!	strong France Europe that defends europe changing crisis

topological clustering of the opionon issued from the tweets, each cluster being associated to a prototype and a weight vector, reflecting the relevance of the data belonging to each clusuter. The proposed framework has been used on a real dataset issued from the tweets collected during the 2012 french election compaign and the experimental results have shown promising performance.

Several perspectives can be considered for this work as: to propose an incremental approach in order to analyze the opinion behavior, to validate the framework on different datasets and to compare the method with the existed methods for the opinion mining.

V. AKNOWLEDGEMENT

This work was supported in part by the AIDEM PEPS HuMains 2013 CNRS Project

REFERENCES

[1] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, no. 1-2, pp. 1–135, Jan. 2008. [Online]. Available: http://dx.doi.org/10.1561/1500000011

[2] B. Li, L. Zhou, S. Feng, and K.-F. Wong, "A unified graph model for sentence-based opinion retrieval," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ser. ACL '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 1367–1375. [Online]. Available: http://dl.acm.org/citation.cfm?id=1858681.1858820

[3] X. Huang and W. B. Croft, "A unified relevance model for opinion retrieval," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ser. CIKM '09. New York, NY, USA: ACM, 2009, pp. 947–956. [Online]. Available: http://doi.acm.org/10.1145/1645953.1646075

[4] B. Liu, *Sentiment Analysis and Opinion Mining*, ser. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2012.

[5] B. Liu, M. Hu, and J. Cheng, "Opinion observer: Analyzing and comparing opinions on the web," in *Proceedings of the 14th International Conference on World Wide Web*, ser. WWW '05. New York, NY, USA: ACM, 2005, pp. 342–351. [Online]. Available: http://doi.acm.org/10.1145/1060745.1060797

[6] S. Gerani, M. J. Carman, and F. Crestani, "Proximity-based opinion retrieval," in *SIGIR*, 2010, pp. 403–410.

[7] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, Nov. 1975. [Online]. Available: http://doi.acm.org/10.1145/361219.361220

[8] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems)*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005.

[9] N. Grozavu, Y. Bennani, and M. Lebbah, "From variable weighting to cluster characterization in topographic unsupervised learning," in *in Proc. Proc. of IJCNN09, International Joint Conference on Neural Network*, 2009.

[10] T. Kohonen, *Self-organizing Maps*. Springer, Berlin, 2001.

[11] C. M. Bishop, M. Svensén, and C. K. I.Williams, "GTM: The generative topographic mapping," *Neural Comput*, vol. 10, no. 1, pp. 215–234, 1998.

[12] M. Lebbah, N. Rogovschi, and Y. Bennani, "BeSOM : Bernoulli on Self Organizing Map," in *IJCNN '07*, Orlando, Florida, 2007.

[13] J. Verbeek, N. Vlassis, and B. Krose, "Self-organizing mixture models," *Neurocomputing*, vol. 63, pp. 99–123, 2005.

[14] H. Frigui and O. Nasraoui, "Unsupervised learning of prototypes and attribute weights," *Pattern Recognition 37(3)*, pp. 567–581, 2004.

[15] C.-Y. Tsai and C.-C. Chiu, "Developing a feature weight self-adjustment mechanism for a K-means clustering algorithm," *Comput. Stat. Data Anal.*, vol. 52, no. 10, pp. 4658–4672, 2008.

[16] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated Variable Weighting in k-Means Type Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27(5), pp. 657–668, 2005.

[17] J. G. Dy and C. E. Brodley, "Feature Selection for Unsupervised Learning," *JMLR*, vol. 5, pp. 845–889, 2004.

[18] S. Guérif, Y. Bennani, and E. Janvier, "μ-SOM : Weighting features during clustering," in *Proceedings of the 5th Workshop On Self-Organizing Maps (WSOM'05)*, Paris 1 Panthéon-Sorbonne University, France, September 2005, pp. 397–404.

[19] M.-H. Huh and Y. B. Lim, "Weighting variables in K-means clustering," *Journal of Applied Statistics*, vol. 36, no. 1, pp. 67–78, 2009.

[20] N. Grozavu and Y. Bennani, "Simultaneous pattern and variable weighting during topological clustering," in *Proceedings of the 18th International Conference on Neural Information Processing - Volume Part I*, ser. ICONIP'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 570–579.

[21] B. Everitt, S. Landau, and M. Leese, *Cluster analysis.* Arnold ; Oxford University Press, May 2001.

[22] P. Hansen and B. Jaumard, "Cluster analysis and mathematical programming," *Math. Program.*, vol. 79, pp. 191–215, 1997.

[23] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[24] A. K. Jain and R. C. Dubes, *Algorithms for clustering data*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988.

[25] F. Bouillot, P. Poncelet, M. Roche, D. Ienco, E. Bigdeli, and S. Matwin, "French presidential elections: What are the most efficient measures for tweets?" in *Proceedings of the First Edition Workshop on Politics, Elections and Data*, ser. PLEAD '12. New York, NY, USA: ACM, 2012, pp. 23–30.

[26] J. Vesanto and E. Alhoniemi, "Clustering of the Self-Organizing Map," *Neural Networks, IEEE Transactions on*, vol. 11, no. 3, pp. 586–600, May 2000.

[27] D. Davies and D. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 1 (4), pp. 224–227, 1974.