SocialSOM: Topic Detection on Twitter by Organizing Tweets on User Similarity

Bernardo Simões 20-25 páginas aproximadamente

Technical University of Lisbon - Taguspark Campus,
Av. Prof. Doutor Aníbal Cavaco Silva — 2744-016 Porto Salvo, Portugal
bernardo.simoes@ist.ul.pt
http://www.ist.utl.pt/en/

Abstract. *70 and at most 150 words, What did I do, in a nutshell?, summarize the paper, should be written last , very short context ,what the objectives of the study were*

Keywords: topic detection, twitter, self-organizing maps, classification, clustering

1 Introduction

With the evolution of social networks websites like Facebook or Twitter throughout the last couple of years, the amount of pertinent content about a specif issue is increasing dramatically, which calls for new ways to make sense and catalog this data. In the other hand find topic sensitive information on social networks is extremely complicated due to the fact that documents have very little content, slang vocabulary and orthographically mistakes or abbreviations.

The value of data presented in sites like Facebook or Twitter as proven its value in papers like “Predicting the future with social media” where it is possible to predict with high precision the value of a movie box office weeks before it debuts.

This project will focus on Topic Detection on Twitter by using a new approach that will not only take in consideration the words in the corpus, but will also take in account the social network to which a tweet belongs to in order to categorize it using the concept of homophily that has been proven in past that is applicable to social networks.

Will be described the objectives of this project, at 2 we will talk about the state of the art solutions related not only to topic detection but also to twitter data analysis and Self-Organizing Maps.In section 4 Architecture of the purposed solution and finally at section 5 it will be discussed how to evaluate results achieved.

1.1 Objectives

The objective of this project is clear, finding topics on Tweets by analyzing their corpus specific characteristics, like number of characters in a tweet, hashtag, "was retweeted", etc.. And contextualize the social network evolving the person that did the tweet.

After characterizing the tweet with information just described, we will use the unsupervised learning clustering technique Self-organizing maps in order to organize the tweets in clusters of topics. Afterwards it will be needed to categorize the clusters in order to know which topic they belong to.

Lastly the resulting topic clusters will be publicly accessible through a website to everybody that visits it.

2 Basic Concepts

In this section we will start by generally describing what Clustering is and how it works at subsection 2.1, then at subsection 3.1 it will be outlined how Self-organizing [12]maps function, which is the Document Clustering algorithm used on this project.

2.1 Document Clustering

Document clustering is an optimal division of data into cathegories without prior knowledge of the data that is being organized, based only on the similarity between documents. Due to the fact that no prior knowledge of the date has to be known Document Clustering is labeled as Unsupervised Machine Learning.

Yuan-Chao Liu et Al [13] described that Document Clustering can be used to a variety of Computer Science fields, such as:

- Natural Language Preprocessing.
- Automatic Summarization.
- User preference mining.
- Improve Text classification results.

In regard to document categorization there are two main types of Document Clustering, Hard Clustering and Soft Clustering. In Hard Clustering one document can only belong to one cluster, while in Soft Clustering one document can belong to multiple clusters.

In regard to document categorization Springorum et al. [18] performed hard and soft clustering with SOMs [12] while identifying polysemous German Propositions. They used regular SOMs to create multiple hard clusters and used Centroid-Based or Preposition-based softening to create Soft Clusters from the Hard Clusters.

The general mathematical description of Document Clustering can be seen in 1 In the first step a data set must be provided in order to cluster the documents.

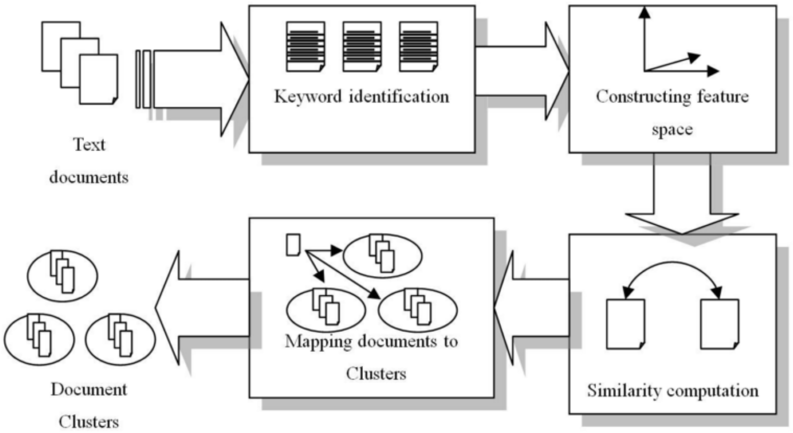

Fig. 1. Text Clustering Main Framework from Dozono [4]

The second step "Keyword identification" is where non relevant words are removed from the documents. Kang [10] proves that keyword removal improves clustering. Another way to extract features is to differentiate text features by analizing the document corpora. For example if the dataset is composed from HTML or XML documents it is possible to identify more relevante features due to the characteristics of the markup. In 3.2 it will be described twitts characteristics as a document and in 4 how feature extraction will be implemented on this project. "Constructing the Feature Space" is characterized by converting the keywords of each document into vectors, the most common algorithm used for this task is SVM (Support Vector Machines). In SVM each vector dimension means one detected key word and each document is represented by the vector of keywords in the feature space. This process and keyword removal is described in Figure 2. Due to the way documents are represented in SVM it is normal that vectors become very large and full of zeros (keywords not present) and it is needed to use sparse vectors to represent the documents in a more efficient way.

Dimensional reduction is done after the construction of the vector space model, in order to reduce the size of the vector space. There are two main ways to do this PCA (Principal Component Analysis) and LSI (Latent Semantic Indexing). PCA calculates the k eigenvectors of the co-variance of the document matrix, which reduces the size of the matrix to k. LSI (Latent Semantic Indexing) works just like PCA but the eigenvectors are calculated directly from the document matrix.

There are two main strategies for Document Clustering, Complete strategie where the data set does not change and Incremental where initial number of document can increase by adding new documents. After a new document is

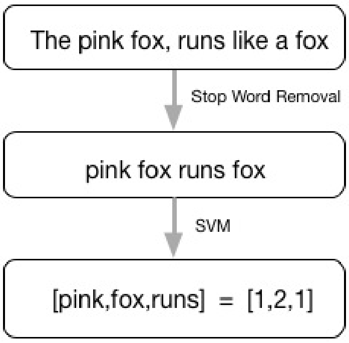

Fig. 2. Stop word removal and transformation to Vector Space Model

added it can be merged into a existing cluster, or can be separated as a new category. While adding new documents it might be needed to re run the clustering algorithm.

After the algorithm converges, cluster similarity can be calculated in multiple ways:

- Shortest Distance Method: Shortest distance between two members of different clusters.
- Longest Distance Method: Longest distance between two members of different clusters.
- Group Average Method: The average distance between all elements of both clusters.
- Centric Method: The distance between the center of two clusters.

There many clustering algorithms, here we will focus on the three most popular.K-means works by randomly selecting k documents as the cluster centroids, then assign each document to the nearest centroid, and finally recalculate the the centroid with new added documents. The algorithm should be executed until convergence which reflects in the centroids stop changing. K-means has the advantage that the number of centroids must be selected before starting the algorithm. AHC or Agglomerative Hierarchical Clustering is hierarquical clustering algorithm where clusters have sub-clusters which have subclusters. Like K-means it is also a simple algorithm that starts by calculating the similarity matrix, then each document is seen as a cluster and finally merge the nearest two clusters into one and update the similarity matrix. The algorithm ends when there is only one cluster or due to clustering entropy.An AHC classic example is species taxonomy where species have subspecies which have subspecies, etc. Lastly there is Self-organizing Maps introduces by [12] which will be used in the thesis and will be detailedly described in the next subsection 3.1.

2.2 The Self-organizing Map

The Self-organinzing map, or for short SOM is a kind of recurrent artificial neural network that as the desired property of topology preservation which mimics the way cortex of high developed animals brains work.

As [2] describes the basic idea behind SOM is to map the data patterns into an n-dimensional grid of neurons or units. That grid is also know as the output space, as opposed to the initial space also called input space, where the input patterns are. Both spaces can be seen in picture 4.

SOMs work similar to the way that is thought that the human brain works by having a set of neurons that through learning experience specialize in the identification of certain types of patterns. These so called neurons are responsible for categorizing input patterns for which they are responsible. Nearby neurons will be organized by similarity which will cause that similar patterns will activate in similar areas of the SOM. With a topology preserving mapping, SOM organizes the information spatially where similar concepts are mapped to adjacent areas. The topology is preserved in a sense that as far as possible neighborhoods are preserved through the mapping process. Neurons are displayed in an n dimensional grid, generally rectangular, but other dimensions are possible like hexagonal or octagonal. The grid of neurons, also called output space can be divided in neighborhoods where neurons responsible for the same kind of input reside. In SOM neurons will have the same amount of coefficients as the input patterns and can be represented as vectors through the SVM model described earlier in section 2.1.

Before describing the algorithm it is important to define three key aspects of the SOM, the learning rate and quantization error. The learning rate is a function that will be decreased in order to converge to zero, it will be applied to winning neurons and their neighbors in order for them to move toward the corresponding input pattern. Quantization Error is the distance between a given input pattern and the associated winning neuron, it describes how well neurons represent the input pattern. The radius of the neighborhood around the winner neuron is particularly relevant to the totpology of the SOM, deeply afecting the unfolding of the output space as stated by [2] The learning phase is characterized by the training algorithm, which works the following way:

- Neurons can be initialized randomly or it is possible to select initialization neurons.
- Given an input pattern, calculate the distance between the input pattern and every neuron on the network.
- The winning neuron will be the closest neuron to the input pattern.
- The neuron will move towards the data pattern at a given learning rate, in order to improve his representation as can be seen in figure 3.
- Neighbor neurons will also improve their representation in order to keep the network progressively organized as can be seen in figure 4.

After the algorithm converges, the prediction phase starts. On the prediction phase new input patterns can be quickly assigned to the SOM, without need to

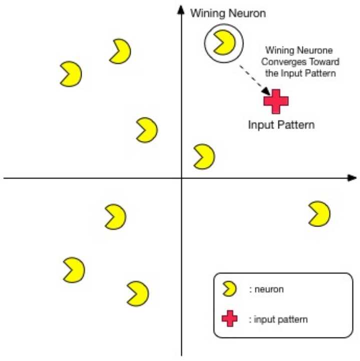

Fig. 3. Winning neuron converging at learning rate

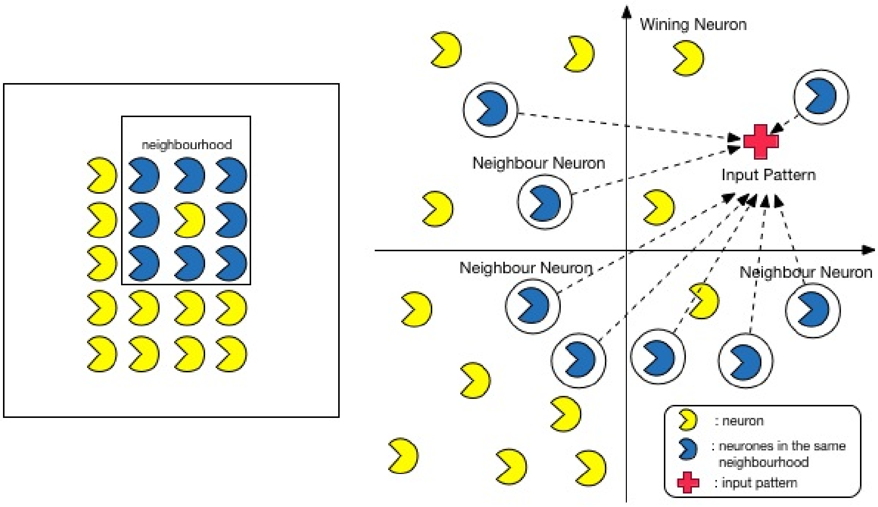

Fig. 4. On the left the output space neighbor, on the right the neighbors of the winning neuron converging

apply the learning rate to the winning neuron and his neighbors (the learning rate converged to zero), it very easy and fast to classify new data now. It is possible that during training the SOM gets stuck while unfolding, this kind of behavior might happen if the input patterns are very complex.

In order to visually interpretate the result of the SOM U-matrices may be used as stated by [2]. The U-matrix is a representation of the SOM in which distances, in the input space between neurons is represented using a gray scale.

The advantages of using SOM is data noisy immunity, easy to visualize the data and parallel processing.

3 Related Work

This section provides insight of work done in multiple research areas that are related to the project. In subsection 3.1 will be described multiple work done using Self-organizing maps. Subsection 3.2 is dedicated to work done on topic detection on the social network Twitter [9]

3.1 Self-organizing Maps

Self-organizing maps are used in a wide are of applications, from authentications systems [4], network intrusion detection [14] and speech recognition and analysis [11].

Detecting Hidden Patterns on Twitter Usage [3] analyzed hidden patterns created buy the natural usage of twitter by its users. In its study they started by collecting data from the twitter API different kinds of topics like "2009 Iran Election" and "iPhone 3.0 OS launch". They made multi level signal extraction not only from information directly present on the tweet, but also by cross referencing with other social website and with the twitter user profile information. The signals retrieved from the social network can be seen in table 1.

Table 1. Twitter Signals

Twitt Corpus	**Twitter Profile**	**External Sources**
Tweet Size	Gender	Other Social Network Accounts
Replies	Number of customizations	Type of website
Re-tweets	Friends to followers ratio	
hashtags	frequency of posts	
Presence of URIs and Type of linked content	Account Age	
Type of Device	Country	
Tweet Location		

By applying clustering algorithm of SOM, they could find 4 demographical clusters during the Iran 2009 Election. The first cluster was characterized by young web-based Iranians, with twitter accounts not older than 3 months with a high frequency of replies. The second cluster was mainly compound of web users from Iran accounts older that 3 months. The third cluster had Iranian users with mobile clients with large texts clearly trying to raise awareness. The fourth and final cluster represented the users around the world trying to raise awareness about the issue by sharing tweets with URIs. Looking at their analysis about the topic ”2009 Iranian Election” it is clear to see that it was possible to describe the type of users represented in the social network and the way they interact with it.

On the iPhone 3.0 OS launch it was possible to find three main clusters. The first cluster was characterized by male users, accounts older than 90 days, coming from countries where the iPhone is marketed, with high adoption of social media clearly representing the target market of the iPhone or its customers. The second cluster had new accounts with higher rate of followers to followees, high frequency of posts per day, presence of URI linking to technology blogs or websites, no country or gender specified meaning that this cluster was clearly composed by news aggregators and technological news websites. Inside the second cluster there was a subcluster of Japanese users which represents the high rate of iPhone adoption in Japan. Finally the third cluster was clearly spammer accounts that where eventually deleted after a couple of months, characterized by popular social connections, posting more than 50 tweets a day with external URIs and the accounts where not older than a day or so.

In conclusion it was possible to detect Twitter usage patterns and specifically detect spammers before they where banned from the social network.

Types of SOMs Depending on the kind of data that scientist are trying to analyze and visualize, different approaches can be made the SOM algorithm in order to better adapt to the data at hand.

Weight adaptation SOMs are simple Self-organizing maps in which the wheights of the vector space model are not even. For example Bação et al. [2] proposed the adaptation of the algorithm in order to better represent geographical data where more weight is given to the coordinates of the data.

First introduced by Ichiki [8], the Hierarchical SOMs are often used when it is possible to decompose on big problem into smaller problems. One or more SOMs are located at each layer usually operating in on different variables. The hierarchical SOM allows the creation of thematic classifications at lower layers which are then composed into a single one Bação et al. [2] which leads to a more specific kind of classification in the lower layers. Based on the survey made by their work Henriques and Lobo [7] suggests that there are four main types of hierarchical SOMs, Thematic Agglomerative, Agglomerative HSOM based on clusters, static devised HSOM and dynamic devised HSOM. Multi layered SOMs where used by Rauber et al. [16] on automatically detect and organize topics in order to organize bookshelves. They used the vector space model to define all

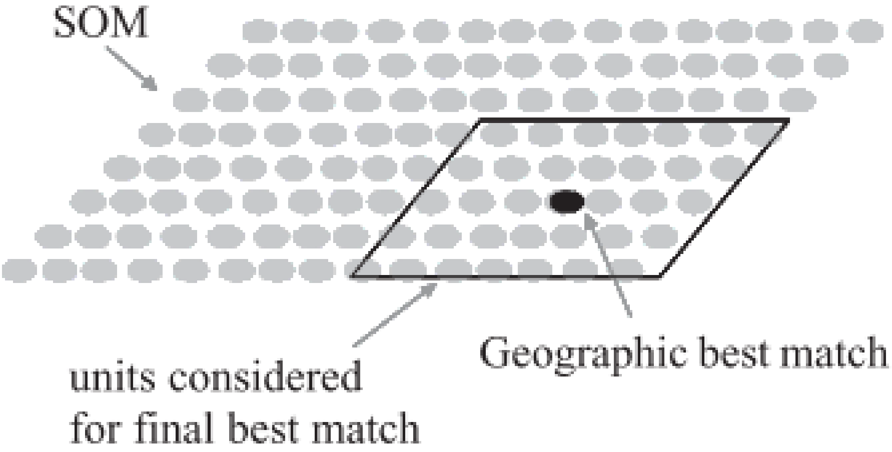

Fig. 5. Geo-SOM structure, from Bação et al. [2]

books and started initially with only four cluster, that where consequently subdivided and categorized which in the end created a hierarchical tree of topics. Multi layered SOMs have been used in a wide variety of applications, such as speech recognition Ichiki [8] and learning to control a robot arm and wrist Sayers [17].

The Geo-SOM Bação et al. [2] applies the first law of geography "Everything is related to everything else, but near things are more related than distant things." to the SOM algorithm, where the winning neuron is chosen by in a radius k defined by the geo-coordinates of the data. In this way the Geo-som forces units that are close in the input space to be close in the output space. The representation of the Geo-som can be seen in figure 5.

Self organizing maps have their own limitations mainly drawing from the fact that it has a fixed number of neurons. Qiang et al. [15] cites in his survey about the state of self organizing maps, that newer algorithms namely Growing cell Structures [6] and Growing Neural Gas [5] don't have this drawback.

3.2 Topic Detection on Twitter

- What did we know about the problem before I did this study?
- What did we do different from previous works?
- Discuss the relevant primary research literature
- Works should be organized by their relevant characteristics
- Comment on why it is relevant for your work
- Comment on what your work does differently

4 Architecture

In this project we are going to apply Self Organizing Maps in order to detect clusters of Topics on Twitter.

4.1 Data Gathering

In order to retrieve data from Twitter, we will be using a ruby library called Twitter Stream [1] , that enables the user to download and inspect the twitter stream of tweets. As the data is gathered it will be stored in a MongoDB database for posterior analyses. As the twitter stream is stored, another function will interact with the twitter API in order to retrieve information from a user profile and relate him with other users by analyzing his followers and who the user is following. In the end of the data-gathering process it will be possible to query the database for:

- Tweets from a user.
- Query tweets for hashtag.
- Query users followers and who he is following.
- Query for tweets that shared the same URI

Tweets will be cathegorized in:

- News Accounts
  - Accounts with a lot of followers
- Profile custumization
- Average number of tweets a day with uri (might suggest spam)

- How am I gonna solve the problem?
- Describe the work that will be done

5 Evaluation Metrics

- How am I gonna evaluate my work?

5.1 Evaluation Criteria by Teachers

- Ability to understand the research problem
- Clear and well defined goals
- Description of the different approaches explored
- Ability to relate the state-of-the-art with the research theme Work methodology and adequate planning for the next stage Organization and quality of the written document
- Inclusion and completeness of updated and appropriate references Oral presentation and discussion

Bibliography

[1] Steve Agalloco. TweetStream, simple Ruby access to Twitter's Streaming API. 2013. URL https://github.com/tweetstream/tweetstream.

[2] Fernando Bação, Victor Lobo, and Marco Painho. The self-organizing map, the Geo-SOM, and relevant variants for geosciences. *Computers & Geosciences*, 31(2):155–163, 2005. ISSN 00983004. doi: 10.1016/j.cageo.2004.06.013. URL http://linkinghub.elsevier.com/retrieve/pii/S0098300404001918.

[3] Marc Cheong and Vincent Lee. A Study on Detecting Patterns in Twitter Intra-topic User and Message Clustering. *2010 20th International Conference on Pattern Recognition*, pages 3125–3128, 2010. doi: 10.1109/ICPR.2010.765. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5597282.

[4] Hiroshi Dozono. Application of Self Organizing Maps to Multi Modal Adaptive Authentication System Using Behavior Biometrics. 2012. URL http://www.intechopen.com/books/applications-of-self-organizing-maps/application-of-self-organizing-maps-to-multi-modal-adaptive-authentication-system-using-behavior-bio.

[5] Bernd Fritzke. Growing cell structures—a self-organizing network for unsupervised and supervised learning. *Neural networks*, 7(9):1441–1460, 1994. URL http://www.sciencedirect.com/science/article/pii/0893608094900914.

[6] Bernd Fritzke. A growing neural gas network learns topologies. *Advances in neural information processing systems*, 1995. URL http://web.cs.swarthmore.edu/~meeden/DevelopmentalRobotics/fritzke95.pdf.

[7] Roberto Henriques and Victor Lobo. Spatial Clustering Using Hierarchical SOM. 2012. URL http://www.intechopen.com/books/applications-of-self-organizing-maps/spatial-clustering-using-hierarchical-som.

[8] H. Ichiki. Self-organizing multilayer semantic maps. 1991. URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=155203.

[9] Evan Williams Biz Stone Jack Dorsey, Noah Glass. Twitter. 2006. URL https://twitter.com/.

[10] Seung-Shik Kang. Keyword-based document clustering. *Proceedings of the sixth international workshop on Information retrieval with Asian languages -*, 11:132–137, 2003. doi: 10.3115/1118935.1118952. URL http://portal.acm.org/citation.cfm?doid=1118935.1118952.

[11] T. Kohonen. The 'neural' phonetic typewriter, pages 11 - 22 . 1988.

[12] T Kohonen. The self-organizing map. *Proceedings of the IEEE*, 1990. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=58325.

[13] YC Liu, Ming Liu, and XL Wang. Application of Self-Organizing Maps in Text Clustering: A Review. 2012. URL http://www.intechopen.

com/books/applications-of-self-organizing-maps/application-of-self-organizing-maps-in-text-clustering-a-review.

[14] Kulkarni P. Nikam D.M. Pachghare, V.K. Intrusion Detection System using Self Organizing Maps. 2009.

[15] Xinji Qiang, Guojian Cheng, and Zhen Li. A survey of some classic self-organizing maps with incremental learning. *2010 2nd International Conference on Signal Processing Systems*, pages V1–804–V1–809, July 2010. doi: 10.1109/ICSPS.2010.5555247. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5555247.

[16] A Rauber, M Dittenbach, and D Merkl. Automatically Detecting and Organizing Documents into Topic Hierarchies: A Neural Network Based Approach to Bookshelf Creation and Arrangement. *Research and Advanced Technology ...*, 135476:1–5, 2000. URL http://medcontent.metapress.com/index/A65RM03P4874243N.pdfhttp://link.springer.com/chapter/10.1007/3-540-45268-0_37.

[17] Craig Sayers. Self Organizing Feature Maps and their Applications to Robotics. *Technical Reports (CIS)*, 1991. URL http://repository.upenn.edu/cgi/viewcontent.cgi?article=1406&context=cis_reports.

[18] Sylvia Springorum, SS im Walde, and Jason Utt. Detecting Polysemy in Hard and Soft Cluster Analyses of German Preposition Vector Spaces. *aclweb.org*, 1998. URL http://www.aclweb.org/anthology/I/I13/I13-1072.pdf.