TÉCNICO LISBOA

HOMOPHILIC SELF ORGANIZING FEATURE MAPS: FINDING TOPICS ON SOCIALY CONNECTED DATA, USING SOCIAL NETWORK RELATIONS

Bernardo Simões

Dissertação para obtenção do Grau de Mestre em

Engenharia de Redes de Comunicações

Júri

Presidente:	Professor Doutor Paulo Jorge Pires Ferreira
Orientador:	Professor Doutor Pável Calado
Vogais:	Doutor whatever full name 2
	Doutor whatever full name 3

Outubro de 2014

Acknowledgments

Remember that your parents have paid for the last 20 something years of your studies and that your advisor had to read this document.

Abstract

Clustering is a widely used technique in data analysis. In this thesis, a generically lgorithm used for clustering is modified in order to enhance the value of socially connected ententies.

To achieve this, we present RubySOM. A framework for easy construction of custom Self-Organizing Maps. With it, it is possible to dinammically change multiple parts of the algorithm, making it extremlly flexible solution to create, train and run custom implementations of the algorithm.

With RubySOM, a relational aware version of the SOM algorithm was created in order to better identify topics on the social network twitter.

Keywords

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

Resumo

Palavras Chave

detecção de tópicos, twitter, mapas auto organizados, classificação, agrupamento

Contents

List of Figures

List of Tables

List of Acronyms

1

Introduction

Contents

With the evolution of social network websites like Facebook and Twitter, the amount of pertinent content about a specif issue is increasing dramatically, which calls for new ways to make sense and catalog this data. The usage of social networks for branding quality and on-line marketing is specially compelling since 19% of all tweets [**?**] and 32% [**?**] of blog posts are about brands or products. On the other hand, finding topic sensitive information on social networks is extremely complicate due to the fact that documents have very little content, slang vocabulary and orthographically mistakes or abbreviations.

?] successfully predicted box-office revenues by monitoring the rate of creation of new topics based on debuting movies. **?**] was able to outperformed some market-based predictors.

Academic and enterprise world is now starting to look at Machine Learning for new ways to achieve revenue and visualize data representing the way the world works. As a consequence, the Machine Learning course at Standford is the one with more students enrolling this year [1] with more than 760 students enrolled.

[**?**] was able to achieved 81.7 percent accuracy in detecting human faces, 76.7 percent accuracy when identifying human body parts and 74.8 percent accuracy when identifying cats. He used a 9-layered locally connected sparse auto-encoder with pooling and local contrast normalization on a large dataset of images (the model has 1 billion connections, the dataset has 10 million 200x200 pixel images downloaded from the Internet) trained using model parallelism and asynchronous SGD on a cluster with 1,000 machines (16,000 cores) during three days. Even though the amount of computing power used in this project was of several order of magnitude, it is remarkable how an unsupervised algorithm could achieve such results.

Even though a lot of solutions have arisen in order to automate real time searches, topic categorization and many other data intensive tasks, Twitter still uses humans in order to deliver ads to trending queries states Edwin Chen's Data Scientist Responsible for ads quality at Twitter. On his blog post [2] Edwin describes the process of Twitter to deliver real time adds to trending queries, the main problems that arise in the Twitter platform in order to identify rising topics are mainly:

- The queries people perform have never before been seen, so it's impossible to know beforehand what they mean.
- Since the spikes in search queries are short-lived, there's only a short window of opportunity to learn what they mean.

This means that when an event happen, people immediately come to Twitter in order to know what is happening in a determined place in real time. Twitter solves this issue by monitoring which queries are currently popular in real time, using a Storm topology [3] and after the queries

[1] http://www.forbes.com/sites/anthonykosner/2013/12/29/why-is-machine-learning-cs-229-the-most-popular-course-at-stanford/

[2] http://blog.echen.me/2013/01/08/improving-twitter-search-with-real-time-human-computation/

[3] http://storm-project.net/

are identified, they are sent to a Thrift API [4] that dispatches the query to Amazon's Mechanical Turk [5] service where real people will be asked a variety of questions about the query.

Social Media Analytics is another raising topic which draws from Social Network Analysis, Machine Learning, Data Mining, Information Retrieval (IR), and Natural Language Processing (NLP). As stated by Melville et al. in [**?**] 32% of the 200 million active bloggers, write about opinions on products and brands. Where 71% of 625 million Internet users read blogs, 78% of respondents put their trust in the opinion of other consumers. In comparison, traditional advertising is only trusted by 57% of consumers. This kind of data drives companies to Social Media Analytics in a way to know what people are saying on the web about their companies and products. This new worry has brought to life a lot of new startups like Sumal[6] or ThoughtBuzz[7] but also solutions from the old players like IBM [8] and SAS [9]

Its also important to notice that in the last few years Data Science/Analysis has been a trending topic, mostly due to the fact that big dot-com companies have been making lots of money through exploiting user specific information in order to deliver ads and sell products. No wonder that if you look that in the top 10 ebooks sold by O'Reilly throughout 2013, four are about data science [10].

In this project we will focus on using an unsupervised learning technique based on neural networks named Self-Organizing Maps [**?**] in order to detect topics in Twitter posts, by using the Social Network users as base neurons for clustering. After the network is trained, it will be possible to categorize tweets in real time. This approach will be better described in subsection **??**.

1.1 Motivation

Given the general description provided previously, what is the motivation of your work. Explain why the product or solution developed in the course of your thesis is important.

1.2 Objectives

The objective of this project is clear: finding topics on Tweets by analyzing their textual specific characteristics, like number of characters in a tweet, hashtags, is a retweeted [11], etc., and contextualize the social network involving the person that did the tweet by retrieving user specific profile information.

[4] http://thrift.apache.org/
[5] https://www.mturk.com/mturk/
[6] https://sumall.com/
[7] http://www.thoughtbuzz.net/
[8] http://www-01.ibm.com/software/analytics/solutions/customer-analytics/social-media-analytics/
[9] http://www.sas.com/software/customer-intelligence/social-media-analytics.html
[10] http://shop.oreilly.com/category/deals/bestoforeillydotd.do?code=DEAL&cmp=twnabooksvideosinfo-authornote_best_of_2013
[11] retweet is when a user shares a tweet that is not his

After building the dataset, it will be needed to train the Self-Organizing map neural network in order to be able to have the network ready for clustering classification of each future tweet that will be added to it. After the SOM training it will be necessary to categorize the clusters in order to know which topic they belong to. When this step ha ended it will be possibly to get new tweets categorized on the moment they enter the network without further delay.

Finally it will be presented a website where a user will be able to login with his twitter account and see all of his tweets being categorization in the moment they are clustered. After all the user tweets are clustered there will a graphical presentation of the user twitter usage where it will be possible to see multiple statistical information such as the topics a user is more interested in and his own twitter Self-Organizing map network of topics with his friends, the wire-frames of the website can be seen in the attachments Section **??** in Figure **??**.

1.3 Contributions

- enhance topic discovery on data with small text corpus and high social significance.
- create a framework to easily edit multiple parts of a SOM algorithm, by passing high order functions as configuration.
- highly customizable infinite tweeter crawler that preserves the social network.

Do not forget this one. Notice that the objectives is what you have proposed to do. Main contributions are the innovations of your work, or, in other cases, what your work is really good at. If you submitted/published an article in a peer-reviewed conference or journal, do not forget to state here.

1.4 Dissertation outline

Explain how did you organized your thesis.

2

Background

Contents

In this Section we will start by generally describing what Clustering is and how it works then we will outline how Self-Organizing maps [**?**] function, which is the Document Clustering algorithm used on this project.

2.1 Document Clustering

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

?] asserted that Document Clustering can be used in a variety of Computer Science fields, such as:

- Natural Language Preprocessing.
- Automatic Summarization.
- User preference mining.
- Improving text classification results.

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 **?**] performed clustering with SOMs [**?**] while identifying polysemous German Propositions. They used regular SOMs to create multiple clusters and used Centroid-Based or Preposition-based softening to create Soft Clusters from the Hard Clusters.

The clustering process usually works as described in 2.1 In the first, step a data set must be provided in order to cluster the documents. The second step is where non relevant words are removed from the documents, which greatly improves clustering quality [**?**]. Another way to extract keywords is to differentiate text features by analyzing the document corpora. For example if the dataset is composed from HTML or XML documents it is possible to identify more relevant features due to the characteristics of the document syntaxe. The fourth step is characterized by converting the keywords of each document into vectors, the most common model used for this task is VSM (Vector Space Model). In VSM, each vector dimension means one detected keyword and each document is represented by the vector of keywords in the feature space. This process an is described in Figure 2.2.

There many clustering algorithms. K-means works by randomly selecting k documents as the cluster centroids, then assigning each document to the nearest centroid, and finally recalculate the centroid with new added documents.

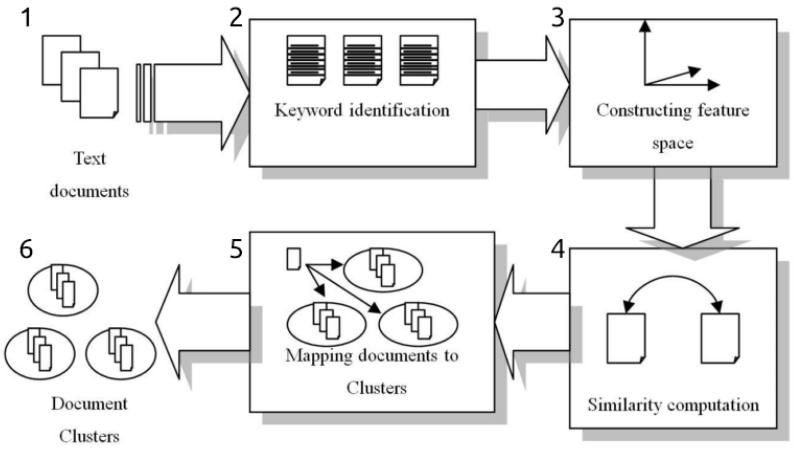

Figure 2.1: Text Clustering Main Framework from **?**]

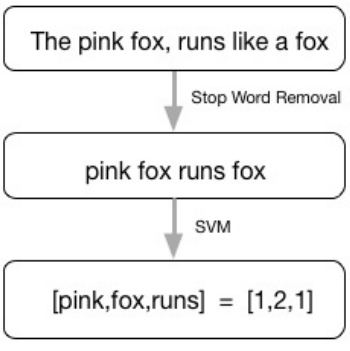

Figure 2.2: Stop word removal and transformation to Vector Space Model

2.2 The Self-Organizing Map

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

Data: Input patterns $X = \{\overrightarrow{x_1},\ldots,\overrightarrow{x_N}\}$
Result: Trainned map and clustered input patterns
for $t = 1\ to\ t_{max}$ **do**
  Randomly draw an input pattern, $\overrightarrow{x_d}$;
  $p = \arg min_i\{\|\overrightarrow{x_d} - \overrightarrow{w_i}\|\}$;
  $\overrightarrow{w_i} = \overrightarrow{w_i} + \epsilon(t) \cdot h_i p(t) \cdot (\overrightarrow{x_d} - \overrightarrow{w_i}), \forall i$
end

Algorithm 1: How to write algorithms

As **?**] 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 Figure 2.4.

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 neurons are responsible for categorizing the input patterns for which they are responsible to identify. Nearby neurons will be organized by similarity which will cause that similar patterns will activate 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 structures are possible, such as 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 VSM model described earlier in Section **??**.

Before describing the algorithm it is important to define two 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 topology of the SOM, deeply affecting the unfolding of the output space as stated by [**?**].

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 a specific initialization.
- 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.

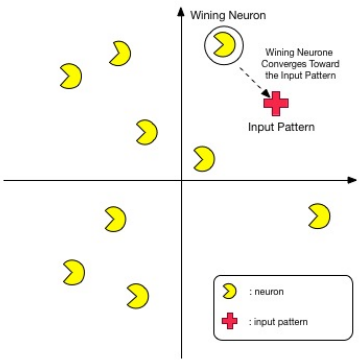

Figure 2.3: Winning neuron converging at learning rate

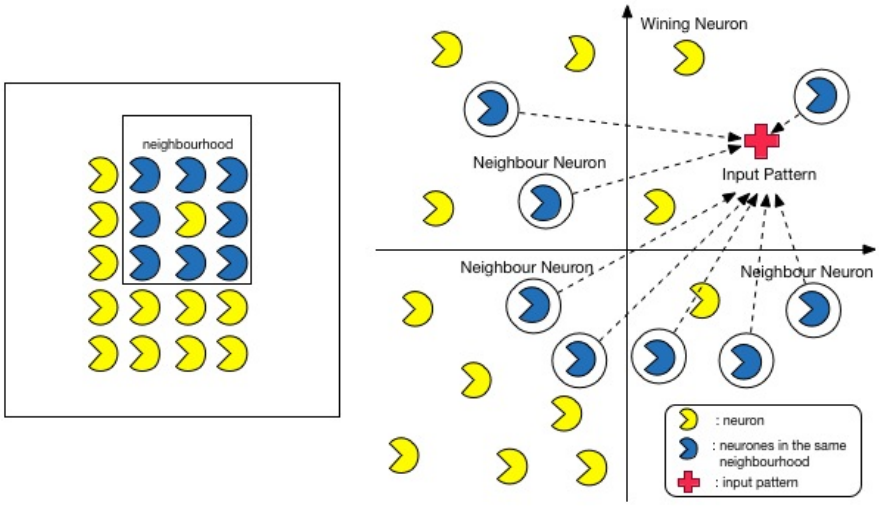

Figure 2.4: On the left the output space neighbor, on the right the neighbors of the winning neuron converging

- 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 2.3.
- Neighbor neurons will also improve their representation in order to keep the network progressively organized as can be seen in Figure 2.4.

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 apply the learning rate to the winning neuron and his neighbors. Thus it very easy and fast to classify new data now.

In order to visually interpret the result of the SOM U-matrices may be used as stated by [**?**]. 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 noise immunity, easy to visualize the data, and parallel

processing [**?**].

3

State of the art

Contents

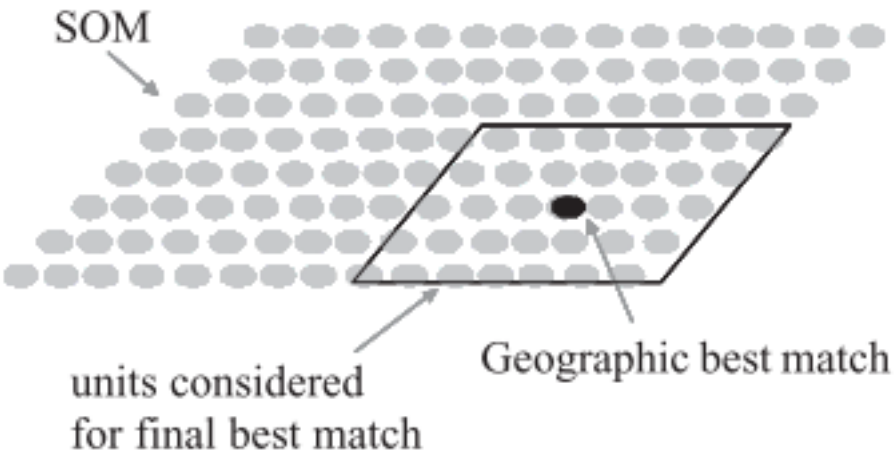

Figure 3.1: Geo-SOM structure, from **?**]

This Section provides insight of work done in multiple research areas that are related to the project. In subsection **??** will be described multiple work done using Self-Organizing maps. Subsection **??** is dedicated to work done on topic detection on the social network Twitter [1]

3.1 Self-Organizing Maps

Self-Organizing maps are used in a wide are of applications, from authentication systems [**?**] through network intrusion detection [**?**] and speech recognition and analysis [**?**].

3.1.1 The Geo-Som

The Geo-SOM **?**] applies the first column 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 in a radius defined by the geo-coordinates of the data, forcing 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 3.1.

3.1.2 Detecting Hidden Patterns on Twitter Usage

Cheon and Lee [**?**] analyzed hidden patterns created the natural usage of twitter by its users. In its study they started by collecting data from the twitter API of 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 3.1.

[1] http://www.twitter.com

Table 3.1: Twitter Signals

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

By applying a SOM, they could find 4 demographical clusters during the Iran 2009 Election. The first columnster 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 columnster 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 sub-cluster 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.

3.2 Topic Detection and Clustering

There have been many topic detection techniques. Many of them rely on the TF ID [**?**] (term frequency – inverse document frequency) which is not particularly adequate for topic detection on Twitter due to the fact that tweets are very small, composed by typos or slang words and might be written in multiple languages, sometimes at the same time. In this subsection we will take a look at multiple methods of topic detection in general and specifically on the Twitter social network.

3.2.0.A Topic and Trending Detection

Due to the rapid adaptation of people to always be on-line, through the usage of cellphones on the move, desktops at work and even theTV at home, the increase of user generated content has increased tremendously in latest years. In 2006 35% of on-line adults and 57% of teenagers created content on the Internet [2], which in "Internet Years" was ages ago. With amount of content increasing, new real-time and scalable algorithms are needed in order to make sense of all this data. **?**] propose a new technique for emerging topic detection that permits real-time retrieval of the most emergent topics expressed by a community on Twitter. Their work applies the PageRank [**?**] algorithm to the users follower/followee relationship in order to find the most influential user on the network, and then calculates the most trending topics by relating, social influence, word co-occurrence and time frame. In the end, an interface was created where it would be possible to navigate hot topics in a given time frame. Topic labeling was not automatic and was implicit by the time frame of an event, if two highly social events would occur in the same time frame with word relations the results could be interpreted as the same, for example if a political candidate would win the elections at the same of an important sports club would win a specific cup, the word win could be trending at the same time for two different topics and due to high temporal dependency they could be interpreted as the same topic. **?**] also used the PageRank algorithm in order to find the most influential twitter users on a certain topic, but uses a different approach where they represent each twitter user as a bag of words comprising of all the tweets that they have posted. Afterwards it uses Latent Dirichlet Allocation [**?**] in order to find the topics each user is interested in. In the end it was possible to prove that follower/followee relation on twitter was not just casual, but that people actually follow other people in which they have some resemblance or common interest. This concept is called homophily and will be further explored by this project.

[2] Data source: http://www.pewinternet.org/Presentations/2006/UserGenerated-Content.aspx

Table 3.2: **?**] tweet characteristics hypotesis versus influence

Hypotheses	Influence of Features
Syntatical	
Tweets that contain hashtags are more likely to be relevant than tweets that don't	Not Important
Tweets that contain an URI are more relevant that tweets that don't	Important
Tweets that are replies to other tweets are less relevant	Important
The longer the tweet is the more relevant it is	Not Important
Semantic	
The more the number of entities the more relevant a tweet is	Important
Different types of entities are of can have different amount of interest to a give topic	Important
The greater the diversity of concepts mentions in a tweet the more likely for it to be relevant	Important
The relevance of a tweet is determined buy its polarity	Important
Contextual	
The lower the temporal distance between a query and the creation of a tweet the more relevant the tweet is	Not Important
The more the number of tweets created by a user the more relevant one of his tweets will be	Not Important

3.3 Twitter Data Mining

In this subsection, we will focus on work done on the Twitter social network in order to leverage insights on how the public data available from the website can correlated within itself and with outside sources.

3.3.1 Tweets Hidden Data

Tweet retrieval and analysis is a double edged problem. On one side the tweet is really small which makes it almost impossible to retrieve any actual sense from it. On the other hand the amount of tweets generated per day is around 140 million [3] wich means that it is very hard to a deep analyses of the semantics and content of individual tweets, and that, only the more appropriate signals should be evaluated. **?**] evaluated how the multiple signals that could be retrieved directly or indirectly from the tweet corpus could mean that a tweet is relevant for a determined topic. In his work, Tao presents premises that seem intuitively true and proves they actually are relevant through a comparison of multiple precision and recall values. Its results on feature comparison where summarized in Table 3.2, the first column consists of all the made hypothesis categorized by type, and the second column tells if the data used actually influenced in precision and recall results. Tau also compared result of topic characteristics, concluding that distinction between local and global events as well as temporal persistence proved to not be relevant on relevance prediction.

?] also approached the issue of having very little content on tweets in order to categorize a

[3] https://blog.twitter.com/2011/numbers

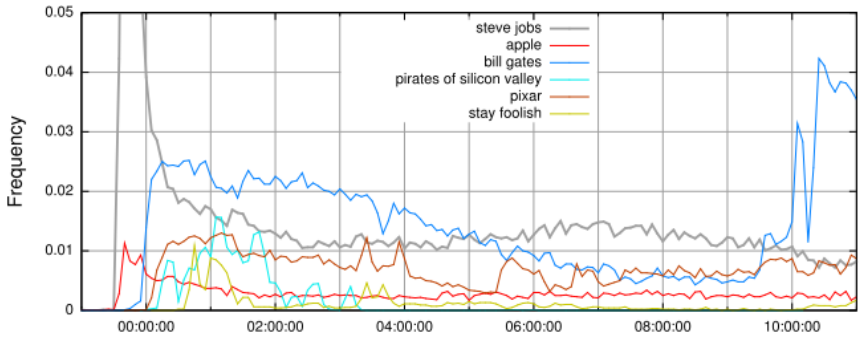

Figure 3.2: The Churn effect: Frequencies of queries related to Steve Jobs death over a 12 hour period in 5-minute intervals, normalized to the total number of queries in the interval. At its peak, the query “steve jobs” reaches 0.15 (15% of the query stream); Graph taken from [**?**]

tweet, and tried to solve it by applying the content of linked URIs into the tweet body in order to improve precision and recall. The best fitting approach was using Field-Based weighting where for each tweet a new document is created which contains two fields; the terms in the tweet and the terms in the linked document. Afterwards a learning to rank algorithm called PL2F is used against the dataset from Microblog2011 in order to find the best weighting that should be applied to the tweet corpus and the URI referenced page. With this trained model they where able to improve precision in an order of 0.9, over only analyzing the text contained in the tweets.

3.3.1.A Rapidly Changing Trends

Due to the real time nature of Twitter, using typical retrieval model that relies on term frequency models like BM25 or language modeling cannot be applied, as stated by **?**]. The study of topic perdurance on the social network proved that it is presented in bursts of queries and mentions of a topic. The typical usage of twitter for search is not the same of Google. When user are searching in twitter they want to find out what is happening right know meaning that classification techniques based on past events cannot respond this kind problem. As stated by **?**] this problem has not yet been solved at Twitter (or anywhere else at the time of writing this report), and issues a new kind of data analysis approach that was not taken into consideration in the past. This effect of rapidly changing topics and queries based on real time events was named ”Churn”, and can be clearly seen in Figure 3.2.

3.4 Summary

Ending section summarizing the chapter is typically a good idea.

Ensure that the next chapter starts in a odd page

4

The work

Contents

- work on the inesc dataset
- work done on the crawller
- work done on RubySOM
- alterations on RubySOM to create Social version of SOM

4.1 Summary

An ending section summarizing the chapter is typically a good idea.

5

Implementation

Contents

```
DefineGlobals
   clock   alias   clk
   reset   alias   rst
   max_latency     17
   feedback        0
   DefineInputs
      X   std_logic_vector(11 downto 0)
   EndInputs
   DefineOutputs
      Y   std_logic_vector(11 downto 0)
   EndOutputs
EndGlobals
```

Figure 5.1: An example code section.

I have no idea what to write here...

Figure 5.1 shows an example of a `\boxedverbatim` section. It allows to put blocks of code within a frame. I think makes a prettier printing.

5.1 Summary

An ending section summarizing the chapter, is typically a good idea.

6

Evaluation Metrics

Contents

Evaluation of the topic detection on Tweets will be made in two distinct ways. The first way will focus on binary classification using the precision and recall metrics, and will be described in Subsection **??**. The second way will focus on statistically testing the SOM learning process and the computed trained network. This testing process will be described in Subsection **??**.

6.1 Testing for Precision and Recall

Precision and Recall are both ways to measure the rate of right guesses made by the trained SOM network, and are defined in the following way:

- **Precision:** Fraction of retrieved instances that where relevant

$$precision = \frac{|relevant\ documents \cap retrieved\ documents|}{retrieved\ documents} \tag{6.1}$$

- **Recall:** Fraction of relevant instances that where retrieved

$$recall = \frac{|relevant\ documents \cap retrieved\ documents|}{relevant\ documents} \tag{6.2}$$

In order to calculate Precision and Recall we need to have the relevant documents and the retrieved documents. The relevant documents are rather hard to determine because they need to be categorized by humans, which is an expensive task.

6.2 Statistically Testing the SOM

SOM training is always subject to some variability due to multiple causes, like the sensitivity of initial conditions, convergence to local minima and sampling variability, as stated by **?**]. This subsection will present statistical tools to measure the quality of the SOM, by measuring its quantization error and topology preservation.

6.2.1 Quantization Error

The SOM Quantization Error is the mean of all Euclidean distances between the observed data points and their corresponding winning neuron. This value might vary depending on the initialization neurons or the order of the input data fed into the SOM while the training is occurring. When applied to an individual input data, represents how well a neuron is representing input data. Since the SOM Quantization Error represents the mean of all quantization errors from all the input data, generally, the lower the error is the best the SOM was trained.
No general formula exists to minimize quantization error [**?**] . What is generally done is just to change the number and values of the starting neurons and the order of the input data in order to train multiple SOMs. In the end the SOM with the lowest quantization error is chosen. In this project since multiple approaches to the SOM algorithm and data representation will be tested, as

described in Section **??**,and the ones having the lower quantization error will be selected for the prototype.

6.2.2 Topology Preservation

The Self-Organizing Map performs a mapping from the n-dimensional input space into the two dimensional output space and where resides one the most fascinating characteristics, which is that the output map tries to preserve the topology from the input space. This grants the SOM algorithm a way to visualize high-dimensional data that other neural networks or clustering algorithms don't have. Even though this is true, sometimes during training it is not possible to preserve the topology of the network. Thus topology preservation can be measured through the Topographic error **?**] which is the proportion of all data vectors for which first and second BMUs [1] are not adjacent units. In this project the Topographic Error will be calculated for all SOM implementations and VSM usages in order to understand if the representation of the SOM output space is well defined.

6.3 Conclusions

[1] unit that is closest to the winning neuron. BMU Best fitting unit

7

Conclusions and Future Work

Draw your conclusions here and sell your work. Trasmit to the juri how hard it was to develop the presented work.

A future work section is usually here.

Appendix A

B

Appendix A