Consistent topics in time inside tweets

Data Mining Course Project Report

Davide Piva
2nd year CS Master Degree @ University of Trento
davide.piva-1@studenti.unitn.it

1 INTRODUCTION AND MOTIVATION

Twitter, as any other social network and blog, is a constant source of unstructured data that, if processed in the right way, can be leveraged to obtain valuable insights in several fields. Due to this reason, lots of companies have started to collect this huge amount of data in order to perform in-depth analysis on it; many of them have initiated to perform sentiment analysis to check customers satisfaction with respect to their products. Other organizations have started to use this asset, hereafter called Big Data, in their decision-making process that includes reporting, exploration of data and exploratory search (e.g., finding correlations). According to upGrad [7], Big Data can help create pioneering breakthroughs for companies that know how to use it correctly.

A challenging aspect in Big Data analysis is to extract valuable insights that can drive business strategies and decisions. In this context, the social network Twitter plays a crucial role during the information retrieval phase: millions of people publish everyday short messages, hereafter tweets, that often contain their desires or demands. The aim of many companies is to leverage these tweets in order to catch common issues or demands to guide their business strategies for their products or services. These insights, according to upGrad [7], can be used to develop new products/services, enhance marketing techniques, optimize customer service, improve employee productivity and find radical ways to expand brand outreach.

In addition, organizations aim also to intercept occasional issues or demands that are not constant over time but are very popular in a bounded period of time. This is the case when there is a sudden issue or request for a particular product or service that affects many customers. Catching this kind of insights, permits companies to better address their business short-term decision-making phase and enhance promptly their products or services. According to Ben Ridler [2], a business owner ability to effectively deal with customer complaints provides a great opportunity to turn dissatisfied customers into active promoters of the business.

Nowadays there are lots of datasets that group together even millions of tweets and many of them are easily and freely accessible by anyone through internet thanks to the fact that there are more than 500 millions new tweets each day [6]. Despite the fact that many of those datasets contain unusable data that force people to pre-process them, companies can elaborate those tweets in order to achieve the above mentioned business goals.

2 RELATED WORK

3 PROBLEM STATEMENT

In order to achieve the goal described in Section 1 (i.e., identify consistent topics in time inside tweets), a public available dataset that groups more than 300.000 tweets that contain the hashtag *#covid19* [1] has been used. As many other datasets composed of unstructured data dumped from a social network such as Twitter, this dataset has lots of fields that characterize each tweet (e.g., publisher's username, location and account information, text of the message, etc.).

To identify consistent topics in time using those tweets, the only fields needed are the publication date of each of them and the relative text. In other words, in the working environment a tweet is defined as a tuple composed by the following fields:

- **date**: the date of the tweet publication, expressed as an integer value that represents the number of seconds that have elapsed since the midnight of 1st January 1970;
- **text**: the text of the tweet represented as a list of words. This list is obtained splitting the source text using the blank character " " as separator.

Therefore, in order to obtain the formal model of this problem, the following sets are been defined:

TWEET = the set of tweets, defined as TIMESTAMP×TEXT where TIMESTAMP $\subset \mathbb{N}_{\geq 0}$ and TEXT $= \{x | x \in \Sigma^*\}$ where Σ is the reference alphabet;

TIMESPAN = the set of all the possible time spans in which the input dataset can be split, defined as a subset of $\mathbb{N}_{\geq 0}$

TS-THRESHOLD = the set of all the possible thresholds to identify frequent terms and topics in a single time span, defined as a subset of $\mathbb{N}_{\geq 0}$

GB-THRESHOLD = the set of all the possible thresholds to identify consistent topics over all the time spans, defined as a subset of $\mathbb{N}_{\geq 0}$

The aim is to model an utility function f defined as

$$f : \text{INPUT} \mapsto \text{TOPIC}$$

where INPUT is defined as

$$\text{TWEET} \times \text{TIMESPAN} \times \text{TSTHRESHOLD} \times \text{GBTHRESHOLD}$$

and TOPIC $= \{x | x \in \Sigma^*\}$ where Σ is the reference alphabet. In other words, TOPIC is the set of all the possible consistent topics over all the identified time spans.

4 SOLUTION

5 IMPLEMENTATION

6 DATASET

As described previously in Section 3, in order to build and test the solution presented in Section 4 a public available CSV dataset with more than 300.000 tweets that contain the hashtag *#covid19* [1] has been downloaded from kaggle.com and processed. As many other datasets that group together huge quantity of unstructured data, it had to undergo a pre-processing phase in order to remove unnecessary features and noisy data.

In that dataset, a tweet is defined as a data item composed of thirteen features: *user-name*, *user-location*, *user-description*, *user-created*, *user-followers*, *users-friends*, *user-favourites*, *user-verified*, *date*, *text*, *hashtags*, *source*, *is-retweet*. In order to achieve the goal described in Section 3, only two features are needed: the publication date of the tweet and the relative text. Due to this reason, the first stage of the pre-processing consists to cut off all the non-relevant data features. This has been achieved quite

easily since the dataset is CSV (Comma Separated Value) file and the Pandas Python library [4] has lots of APIs to manage efficiently such big datasets.

After this initial stage, each tweet has been processed singularly in a multi-process execution where to each process is assigned a portion of the input dataset. For each tweet in the dataset portion, each process has to manipulate both the date and the text fields in the following manner:

Date field is transformed into an UNIX timestamp. In particular the publication date is stored in the input dataset as a string formatted as "yyyy-mm-dd hh:mm:ss", but thanks to the *datetime* Python library this transformation can be performed in a easy and fast way. With the aim to be complaint also with the format of another dataset that groups together tweets in order to perform sentiment analysis [5], the script responsible for the date manipulation is capable to transform also dates formatted like "Mon Apr 06 22:19:45 PDT 2009" into a UNIX timestamp.

Text field is transformed into a list of words. In particular, thanks to the *nltk* [3] Python library, the initial text has been split using the blank character " " as separator and only the useful words that can be leveraged in order to derive a topic have been kept. Due to this reason, only nouns and adjectives appear in the final result list. Furthermore, there are other situations where a word, even it is a name or an adjective, cannot be considered:

- if the word contains a slash "/", non-ASCII characters, special Unicode sequences;
- if the word represents a numeric value;
- if the word is composed by only one character;
- if the word is a stop word. This method has been implemented because there are some words like "https" or "amp" that are tagged as nouns but they cannot be leveraged to build a topic.

As a note, if the preprocessor script has been executed in the debug mode, all the non-considerable words are dumped inside a separated CSV file (together with the relative timestamp) in order to check if the script cut off useful words. Then, each considerable word is filtered and sanitized in order to remove all the eventual noisy characters. In particular, the filtering method performs the following operations:

(1) all the word's characters are lowered;
(2) all the emojis are removed;
(3) all the non-alphanumeric characters are substituted with a blank space (e.g., the word "white-house" is transformed to "white house");
(4) the word is split again using the blank character as separator to identify eventual other words after the previous operation;
(5) all the obtained words are rechecked again to see if they are considerable and, eventually, discarded;
(6) as a final operation, each of the obtained words are checked with an aliases map. If there is an entry for a word, then the associated value is substituted. The aim is to generalize as much as possible the words used inside the tweets leveraging the knowledge about that dataset in order to obtain better results: since all the texts contain the word "covid19" or one of its many variations (e.g., "sars-cov-2", "covid", "coronavirus", etc.), all of them are mapped to the word "covid19".

After these filtering operations, a final check is performed over all the considerable words in order to find out if there are doubled words or empty string.

When each process has terminated its execution, all the partial results are collected in a single list by the master process using a