Filtering Noisy Web Data by Identifying and Leveraging Users' Contributions

Alina Stoica
EDF R&D
1 av. du General de Gaulle,
92140 Clamart, France
alina.stoica@edf.fr

Abstract

In this paper we present several methods for collecting Web textual contents and filtering noisy data. We show that knowing which user publishes which contents can contribute to detecting noise. We begin by collecting data from two forums and from Twitter. For the forums, we extract the meaningful information from each discussion (texts of question and answers, IDs of users, date). For the Twitter dataset, we first detect tweets with very similar texts, which helps avoiding redundancy in further analysis. Also, this leads us to clusters of tweets that can be used in the same way as the forum discussions: they can be modeled by bipartite graphs. The analysis of nodes of the resulting graphs shows that network structure and content type (noisy or relevant) are not independent, so network studying can help in filtering noise.

Introduction

For services or utility providers such as EDF (Electricité de France, one of the biggest energy suppliers in the world), it is very important to be aware of their customers' opinions on the different services and products they propose. The Internet represents an important source of information since clients often use Web social media to ask questions, discuss news and express opinions. It is thus essential to be able to collect, filter and analyze Web contents.

In this paper we present a set of methods we have used in order to collect and prepare Web textual contents related to EDF for further analysis (like text mining or qualitative analysis). In our opinion, an important step is the filtering of noise as noisy contents can bias the analysis. We show that using the relations present in the data (which user publishes which contents) can help filtering noise in addition to classical strategies. Our data sources were the social network, Twitter, and two forums: droitFinances (http://droit-finances.commentcamarche.net/forum/) and the French version of yahooQA (http://fr.answers.yahoo.com/).

Twitter has been intensely used since its creation for information diffusion. Even if in France it is less popular than in the United States, for instance, the number of French accounts has been continually growing since its creation, and was estimated at 2.4 million in March 2011[1]. In our opinion, Twitter represents an important source for learning how news about a company is understood and welcome.

If people widely use Twitter for information diffusion, they generally use discussion forums in order to ask questions, search for answers and advice and discuss their problems. By reading Google alerts on several words related to EDF's customer relationship (e.g. the French translation of ”electricity bill”, ”EDF problem”) from April 2010 to October 2010, we learned that EDF customers mainly talked about their relations with EDF on forums. What's more, 74% of the contents belonged to two forums: droitFinances and yahooQA. We therefore decided to search textual contents about EDF on these two forums and on Twitter.

Data collection

As we wanted to collect all texts concerning EDF, whatever the topic, we simply used the key-word ”EDF” in our research of contents.

Twitter data. We used the Twitter API in order to collect tweets written in French containing the word ”EDF” from March 8th 2011 to June 24th 2011. We thus downloaded 23574 tweets.

Forum data. On each forum, we used the internal search engine in order to find all the available discussions where the question or the answers contained the word ”EDF”. We downloaded each discussion and extracted the meaningful information i.e. the author, date and text of each post. We also recorded the section of the forum where the question was asked. The resulting datasets contain 916 discussions in the case of droitFinances and 955 in the case of yahooQA.

As we downloaded all contents containing the word ”EDF”, the collected data contained noise i.e. contents that are not relevant for EDF. Probably a more specific request (e.g. ”nuclear accident Fukushima”) would lead to results that are indeed related to the given topic. However with a specific request one may miss interesting contents that do not contain the words in the request.

Noisy contents must be filtered from the datasets because they can lead to false analysis: false statistics on users' interest for the given subject (here the company EDF), false conclusions on the opinions etc.

Understanding the source of noise is the first step in the process of filtering. In our case, some noise is caused by the fact that ”EDF” stands for ”Electricité de France” but also

[1] http://semiocast.com/publications/

for ”équipe de France” (French for ”French team”). Twitter users tend to write the shorter word ”EDF” when they talk about a French team as tweets are limited to 140 characters. To filter such noisy tweets, we defined a ”black list” of words related to sports (name of sports, teams, players, coaches etc.) and eliminated from the dataset all tweets that contained them. Although by this operation we manage to filter many tweets on sports, it is impossible to have a complete black list; it would mean knowing all the names of all possible sports, players, teams, coaches etc.

Besides contents on sports, our datasets also include other contents that do not strictly speaking concern EDF. For instance, EDF bills can be used in France as a proof of domicile, so discussions about creating bank accounts may get collected in our datasets, even though they are not relevant for us. In order to filter such discussions from the forums datasets, we only kept discussions where the questions contained one of the words: ”EDF”, ”électricité” (French for ”electricity”), ”bleu ciel” (EDF’s trade mark) and some variants (that can be generated by spelling errors). It is unlikely that someone asks a question concerning EDF without mentioning any of these words. We also hand-selected the sections of the two forums that may contain discussions related to EDF and eliminated the discussions not belonging to these sections. For instance, a section called ”consumption” can be useful, while sections called ”banks and accounts” or ”football” are unlikely to contain discussions of interest to EDF.

By applying these simple filters, the data is greatly reduced: the resulting droitFinances dataset contains 47% of the initially downloaded discussions, the yahooQA dataset 25% and the Twitter dataset 36%. This shows the importance of filtering noise. Analyzing the collected data directly would surely lead to false results. In spite of all the filters, some noise is still present. As we will show in the following, we can filter out other noisy contents by using the relations present in the data. But before that, we need to discuss tweets clustering.

Tweets clustering

Besides filtering noisy tweets, it is also useful to identify almost identical tweets before performing any analysis. The presence of tweets with almost the same text is caused by several factors. First, a retweet has often the same text as the original tweet, possibly with ”RT@author” added at the beginning. Second, many online media (i.e. journals) propose shortcuts that allow users to send a tweet containing the title of an article. If several visitors of the site use the shortcut of the same article, the corresponding tweets contain the same text. Third, people may hear a special phrase or title of a report on the television, for instance, and write a tweet reproducing it exactly. In this case too, the corresponding tweets have (almost) the same text.

We want to identify and cluster such tweets with very similar text that reproduce the same content (a tweet or a title or a phrase). Our goal here is similar to that of (Leskovec, Backstrom, and Kleinberg 2009) where the authors detect textual variants of the same phrase in a blog dataset using graph techniques. We, however, do not want to use a minimal length or frequency for the phrases (in our case tweets) we try to identify. Tweets must be clustered if their contents are very similar, no matter how long or, rather, short they are, or how often their words appear in the dataset.

We perform a hierarchical clustering of tweets. For that, we have to define a distance between tweets based on their texts. We begin by defining the *set of words* of a tweet as the set of all the words in its text, except words prefixed by ”RT@”, ”@” and ”#” (along with the prefixes), URLs and common empty words. We then define the distance between two tweets as the *Jaccard distance*[2] between their sets of words.

After several tests and hand evaluation of resulting clusters, we choose to limit the distance between any two tweets of the same cluster at 0.5. We thus obtain 4618 clusters among which 3405 contain only one tweet. The largest cluster contains 144 tweets.

The clustering of tweets is useful because it reduces the quantity of contents to analyze: as tweets in a cluster are practically copies of the same text, one can analyze only one tweet per cluster. This way text mining or qualitative analysis are not overloaded unnecessarily. What’s more, tweets clustering can help in filtering out noisy contents, as we show in the following section.

Filtering noisy data with graph algorithms

In this section, we leverage the existing relations in the datasets (i.e. which users contribute to which discussion / cluster of tweets) for evaluating and filtering contents.

For each forum dataset, we define a bipartite graph where the vertices correspond to the discussions and to their authors[3]. We put a link between a vertex d corresponding to a discussion and a vertex a corresponding to an author if a wrote a comment in d. For the Twitter dataset, we define a bipartite graph in a similar way using the clusters of tweets and their authors[4].

Our purpose is to analyze the three bipartite graphs and see if noisy, non-relevant discussions / clusters have ”special positions” in the graphs.

We make the following assumptions that we try to test subsequently:

- Relevant discussions and clusters belong principally to the main connected component (i.e. the component with the greatest number of nodes) and are ”well connected” to the graph;
- Conversely, the other connected components contain non-relevant discussions or clusters;
- The ”periphery” communities in the main connected component may contain non-relevant discussions or clusters.

These assumptions are based on the idea that most discussions and clusters are relevant. As they concern the same

[2] For two sets A and B, the Jaccard distance is $1 - \frac{|A \cap B|}{|A \cup B|}$.

[3] We call *author* of a discussion a person who wrote a comment in the discussion but who did not ask the question.

[4] We call *author* of a cluster of tweets a person who published a tweet in the cluster.

topic, EDF, they probably share authors. Discussions or clusters that do not have any authors in common with any part of the main component may be related to other topics. Also, ”periphery” communities of authors and their discussions / clusters may correspond to special topics as they are groups poorly connected to the others.

To evaluate the ”well connected” and ”periphery” status, we use ranks inspired by the core - periphery definition for unweighted graphs (Wasserman and Faust 1994). Given a graph $G = (V, E)$[5] and a function $f : V \to \mathbb{R}$, we compute the ranks $r : V \to \mathbb{N}$ using Algorithm1.

Algorithm 1 ***computes ranks of vertices.***

```
Input: A graph $G = (V, E)$ and a function $f : V \to \mathbb{R}$
Output: A function $r : V \to \mathbb{N}$
1. for all $v \in V$, let $r(v) = -1$
2. let $i = 0$
3. while $\exists v \in V$ s.t. $r(v) = -1$, repeat:
   3.1. compute $f(v)$ for all $v \in V$
   3.2.for all $v \in \arg\min_V(f)$ do:
        $r(v) = i$
        remove $v$ from $V$ and all its edges from $E$
   3.3. $i \leftarrow i + 1$
```

This algorithm attributes a small rank to vertices that have a small value for the function f, and a great rank to vertices with a great value of f. What's more, the algorithm takes into account not only the vertices themselves, but also the vertices to which they are connected (as vertices are eliminated iteratively and f is reevaluated on the remaining ones).

In case of a *weighted graph*, we define f for each vertex as the sum of the weights of its links (we denote this function by f_w). By this definition, vertices with small ranks are poorly connected to the graph: they have few links and their links have low weights.

In case of a *bipartite graph* $G = (V, E)$, we denote the function f by f_b defined as $f_b(v) = d_1(v) \times m + d_2(v)$ for all $v \in V$, where $d_1(v)$ is the number of neighbors of v, $d_2(v)$ is the number of vertices at distance 2 from v (i.e. vertices that can be reached in exactly 2 hops from v) and m is a constant greater than all d_2 of all vertices. This definition leads to the elimination in the first steps of vertices that have few neighbors; for the same number of neighbors, the eliminated vertices are those that share neighbors with few vertices from their set (as G is a bipartite graph, V is composed of two disjoint sets).

Graph characterization versus noisy attribute

We propose to characterize the vertices of our bipartite graphs using the following computations:

1. compute the connected components of the bipartite graph;
2. compute ranks of nodes in the main connected component using the function f_b;
3. for the main component, compute communities with the Louvain method (Blondel et al. 2008);
4. compute ranks of vertices in the communities graph using the function f_w.

Let us explain the last point. We define a weighted graph (called *communities graph*) such that the vertices are the communities in the main component and the weighted links correspond to the number of links, in the bipartite graph, between the vertices of the communities.

Vertices of the original bipartite graphs are thus characterized by the type of component to which they belong (isolated or main) and, in the case of vertices in the main connected component, by their rank and the rank of their community. We want to see if this characterization is correlated to the class (noise or relevant) of forums discussions and clusters of tweets. We begin with the first parameter, the type of component, and check the isolated components.

We are interested in isolated components with at least two discussions or clusters. The other isolated components do not share any authors with other components, so we consider them too special. There are only two components in the case of the forum droitFinances and three in the case of yahooQA with at least two discussions. By reading the corresponding discussions, we observe that they talk about other energy providers or about technical matters on heat pumps or solar panels, so none of these discussions strictly speaking concern EDF.

In the case of Twitter, there are 368 components containing at least two clusters. Among them, only 66 have at least two authors. Every other isolated component with more than one cluster represents the activity of only one user, who never takes part in clusters with other authors. We have read the tweets of these 66 components with some ”minimal activity”. 53% of the clusters in isolated components are about sport, 10% contain advertisements for jobs at EDF and 4% are written in English. Another 14% of the clusters contain advertisements for house rents and scholarships payed by EDF. To sum up, 81% of the clusters in isolated components that have ”some activity” (at least two authors and two clusters) do not strictly speaking concern EDF.

Let us now focus on the vertices belonging to the main component. For each forum, we read all the discussions whose corresponding vertices belong to the main component, and labeled them as ”relevant” or ”noise”.

Figure 1 presents the discussions of the two forums function of the two ranks. We observe that most noisy discussions are situated on the left or bottom side of the images. When a discussion belongs to a low-connected community (so a ”specific” community, whose rank is low), it has a high probability of being non-relevant, except if it has a high rank. When the discussion belongs to a high-rank community, it has a good probability of being relevant, except if its rank is low.

Besides the two ranks, there are a lot of other notions that measure how well a node is connected to a graph or how central it is. The closeness and the betweenness centralities, the clustering coefficient, the local description with small patterns (Stoica and Prieur 2009), the page rank are only some examples of such parameters; a survey can be found for instance in (Wasserman and Faust 1994) or (Chakrabarti

[5] V is the set of vertices of G and E is the set of edges.

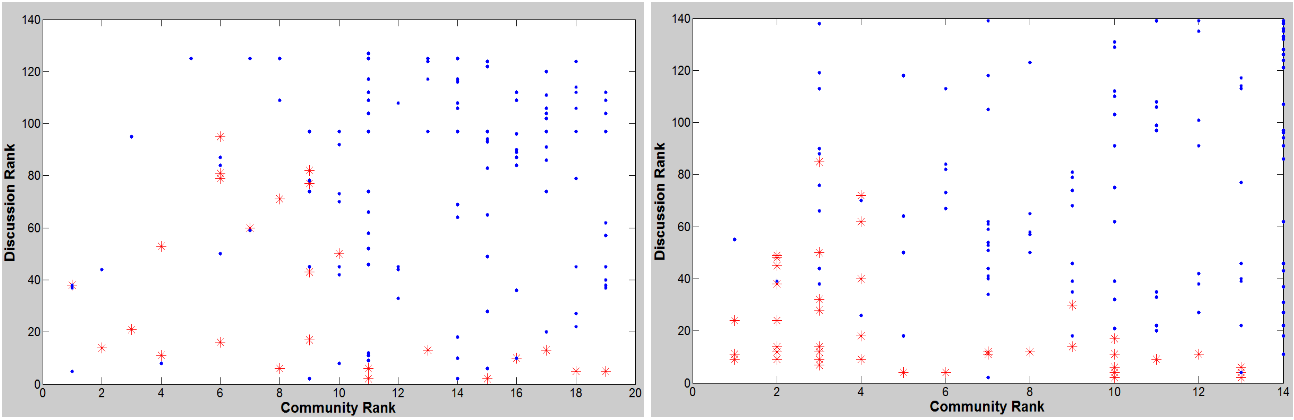

Figure 1: The relevant (blue dots) and noisy (red stars) discussions of the forums droitFinances (left) and yahooQA (right) function of the two ranks.

and Faloutsos 2006) and in (Latapy, Magnien, and Vecchio 2008) for bipartite networks.

We have checked the link between the labels of the discussions and several of these notions. For all of them, including the two ranks, we first performed an ANOVA test (Scheffé 1959) with the null hypothesis that the average value of the parameter is the same for the two classes of discussions. The null hypothesis is rejected for the two ranks.

To characterize discussions (represented by nodes in the bipartite graph), we used the degree (i.e. the number of authors), the number of neighbors with degrees superior to 1 (i.e. ”active” authors who took part in other discussions), the betweenness and the closeness centrality. The null hypothesis is rejected only for the closeness centrality, so there does not seem to be a link between the other parameters and the class of the discussions.

For each community (represented by a node in the weighted graph), we computed the number of links inside the community, the number of links connecting nodes of the community to nodes of other communities, the betweenness and the closeness centrality, the number of triangles containing the node and the sum of weights of links of the triangles containing the node. The null hypothesis is rejected for all these parameters.

Next, we built decision trees using the two ranks and all the above parameters for which the null hypothesis was rejected. The resulting trees do not take into account the number of triangles, their sum of weights and the betweenness centrality of communities, so these parameters are not discriminant. The trees work slightly better than the decision trees based on the two ranks only. The number of wrong guesses of the trees using all the parameters and the two ranks is 9 (out of 373) for the droitFinances dataset and 8 (out of 182) for the yahooQA dataset. Without the two ranks, the number of wrong guesses is 15 and 13 respectively. When using only the two ranks, the number of wrong guesses is 10 for the two datasets. Even if the differences are small, the two ranks seem to be the best choice (given this set of parameters in any case) for characterizing discussions.

Conclusions

In this paper, we presented several methods we used for collecting Web textual contents and filtering noisy data. In addition to classical strategies to filter noise, we used graph modelings to characterize forum discussions and tweets. Many noisy discussions and tweets proved to belong to isolated components. For the discussions in the main connected component, we tried to explain their relevance with several network notions. Among them, the two ranks seem to be the best compromise between number of parameters and accuracy.

The graphs we used are based on the relations present in the data: which user takes part in which discussion / cluster. The clustering of tweets, initially developed to identify ”copies” of the same text and thus avoid redundant analysis, served to make the necessary link between users and their publications.

References

Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* (10):P10008 (12pp).

Chakrabarti, D., and Faloutsos, C. 2006. Graph mining: Laws, generators, and algorithms. *ACM Computing Surveys (CSUR)* 38(1).

Latapy, M.; Magnien, C.; and Vecchio, N. D. 2008. Basic notions for the analysis of large two-mode networks. *Social Networks* 30(1):31 – 48.

Leskovec, J.; Backstrom, L.; and Kleinberg, J. 2009. Memetracking and the dynamics of the news cycle. In *KDD*.

Scheffé, H. 1959. *The Analysis of Variance*. New York: John Wiley & Sons.

Stoica, A., and Prieur, C. 2009. Structure of neighborhoods in a large social network. In *CSE (4)*, 26–33. IEEE Computer Society.

Wasserman, S., and Faust, K. 1994. *Social Network Analysis: Methods and Applications.* Cambridge University Press.