Detecting Real-World Influence Through Twitter

Jean-Valère Cossu[*], Nicolas Dugué[†], Vincent Labatut[*]

[*] Université d'Avignon, LIA EA 4128, France
[†] Université d'Orléans, INSA Centre Val de Loire, LIFO EA 4022, France
Emails: {jean-valere.cossu, vincent.labatut}@univ-avignon.fr, nicolas.dugue@univ-orleans.fr

Abstract—In this paper, we investigate the issue of detecting the *real-life* influence of people based on their Twitter account. We propose an overview of common Twitter features used to characterize such accounts and their activity, and show that these are inefficient in this context. In particular, retweets and followers numbers, and Klout score are not relevant to our analysis. We thus propose several Machine Learning approaches based on Natural Language Processing and Social Network Analysis to label Twitter users as *Influencers* or not. We also rank them according to a predicted influence level. Our proposals are evaluated over the CLEF RepLab 2014 dataset, and outmatch state-of-the-art ranking methods.

I. INTRODUCTION

Social Media have become a wonderful outlet for self-reporting on live events, and for people to share viewpoints regarding a variety of topics. The real-time and personal nature of Social Media content makes it a proxy for public opinion and a source for e-Reputation tracking. Among those media, the most popular one is arguably Twitter.

Twitter. This online *micro-blogging* service allows to publicly discuss largely publicized as well as everyday-life events [1] by using *tweets*, short messages of at most 140 characters. To be able to see the tweets posted by other users, one has to subscribe to these users. If user u subscribes to user v, then u is called a *follower* of v, whereas v is a *followee* of u. Each user can *retweet* other users' tweets to share these tweets with her followers, or mark her agreement [2]. Users can also explicitly *mention* other users to drag their attention by adding an expression of the form `@UserName` in their tweets. One can reply to a user when she is mentioned. Another important Twitter feature is the possibility to tag *tweets* with key words called *hashtags*, which are strings marked by a leading sharp (#) character.

According to Myers *et al*. [3], in 2012, the 175 million active users were connected by roughly 20 billion subscriptions. In 2015, Twitter counts 284 million monthly active users [4]. As we can see, Twitter is now a very widespread tool. Thus, celebrities such as Britney Spears [5], Barack Obama [6], [5] during his presidential campaign, and some organizations [7] largely base their communication on Twitter, trying to become as *visible* and *influential* as possible.

Influence. The *Oxford Dictionary* defines influence as *"The capacity to have an effect on the character, development, or behavior of someone or something"*. Various factors may be taken into account to measure the influence of Twitter users. As described in Section II, most of the existing academic works consider the way the user is interacting with others (e.g. number of followers, mentions, etc.), the information available on her profile (age, user name, etc.) and the content she produces (number of tweets posted, textual nature of the tweets, etc). Several influence assessment tools were also proposed by companies such as Klout [8] and Kred [9]. The ways they process their influence scores are kept secret, and can therefore not be discussed precisely, however they are known to be mainly based on interactions [10].

Interestingly, these tools can be fooled by users implementing simple strategies. Messias *et al*. [11] showed that a *bot* can easily appear as influential to Klout and Kred. Additionally, Danisch *et al*. [10] observed that some users called *Social Capitalists* are also considered as influential although they do not produce any relevant content. Indeed, the strategy applied by *social capitalists* basically consists in following and retweeting massively each other. On a related note, Lee *et al*. [12] also showed that users they call *Crowdturfers* use human-powered crowdsourcing to obtain retweets and followers. Finally, several data mining approaches were proposed regarding how to be retweeted or mentioned in order to gain visibility and influence [13], [14], [15], [16].

A related question is to know how the user influence measured on Twitter (or some other online networking service) translates in terms of real-world, or more precisely *offline* influence. Some researchers proposed methods to detect *Influencers* on the network, however except for some rare cases of very well known influential people, validation remains rarely possible. Thus, there is only a limited number of studies linking real life (offline) and network-based (online) influence. Bond *et al*. [17] explored this question for Facebook, with their large-scale study about the influence of friends regarding elections, and especially abstention. They showed in particular that people who know that their Facebook friends voted are more likely to vote themselves. More recently, two conference tasks were proposed in order to investigate *real-life* influencers based on Twitter, see PAN [18] and RepLab [19] overviews for more details.

Contributions. In this paper, we perform a comparative study of Twitter-based features allowing to measure the offline influence of a user. In other words, we investigate for specific Twitter characteristics that can describe people known to be influential *in real-life*. To answer this question, we conduct experiments on the CLEF RepLab 2014 dataset[1], which contains Twitter data including influence-annotated Twitter profiles. We

---

[1]Data publicly available at http://nlp.uned.es/replab2014/

take advantage of these manual annotations to train several Machine Learning (ML) tools and assess their performance on classification and ranking issues. The former consists in determining if a user is influential or non-influential, whereas the latter aims at ranking users depending on their estimated influence level.

Our first contribution is to review the most widespread Twitter-based features used for user profile characterization problems. In particular, we *simultaneously* consider features traditionally used *separately* by researchers from the Social Network Analysis (SNA) and Natural Language Processing (NLP) domains; and additionally propose a few new features. Our second contribution is the systematic assessment of these features, relatively to the prediction of real-life influence. We show that most features behave rather poorly, and discuss the questions raised by this observation. Finally, we describe two NLP ranking methods that give better results than known state-of-the-art approaches.

The rest of this paper is organized as follows: the next Section reviews the main recent works related to the characterization of Twitter users, in particular in terms of influence. We then describe the RepLab 2014 task in Section III, focusing on the dataset, the evaluation methods, and the results obtained during the campaign. In Section IV, we describe the features selected and defined for our experiments. In Section V, we present our methods and the results we obtained. Finally, we highlight the main aspects of our work in Section VI, and give some perspectives.

II. RELATED WORK

Many works have been dedicated to the characterization of Twitter profiles, which is a problem relatively close to that of detecting influential users. Indeed, the latter can be seen as a specific case of the former. Thus, we included both types of work in this review. Moreover, we distinguished work from the SNA and NLP domains. Some of the features they use are similar, but the former generally put the focus on the fields constituting the profiles and on the way users are interconnected, whereas the latter prefer to use the tweets textual content.

A. SNA Works

Danisch *et al.* [10] showed it is possible to distinguish between different Twitter account behavior using meta-data associated to accounts. In particular, they considered profile data, clients used to tweet, stylistic aspects of tweets, local topology and some tweets characteristics (Table I). Then, using these features, they trained classifier to discriminate regular users from *social capitalists*. They showed that classifiers such as logistic regression and random forests gives highly reliable results. Lee *et al.* [20] also showed, with a study focused on spammers, that these kinds of features are highly relevant to distinguish spammers from real users using the same classification algorithms.

Regarding influence itself, most existing works consider the quantity of followers and the amount of interactions, i.e. the numbers of retweets and mentions. The Klout [8] algorithm is kept secret, but we however know that it is also based on interactions [10]. Intuitively, the more a user is followed, mentioned and retweeted, the more he seems influential [21]. Nevertheless, there is no consensus regarding which features are the most relevant. For instance, Weng *et al.*[22] proposed a modification of the PageRank algorithm and thus focus on the followers, whereas Anger & Kittl take only the interactions into account by using ratios called *Followees/Followers*, *Retweet and Mention* and *Interactions* [23]. The *Retweet and Mention* ratio is the fraction of tweets leading to a retweet or a mention. The *Interactions* ratio considers the distinct number of users that retweeted or mentioned a user divided by her number of followers. Anger & Kittl [23] defined the *Social Networking Potential* of a Twitter user as the mean of these two ratios and used it to rank users.

B. NLP Works

In the NLP domain also, many researches consider various features to characterize Twitter users. The description of the Author Profiling task at CLEF PAN 2014 [18] provides a nice overview of the recent progress in this area. PAN participants investigated various text pre-processings, removing URLs, user mentions and hashtags from tweet contents. They also considered *Stylistic Features* deducted from the tweets content, well known in Information Retrieval (punctuation signs frequencies, average numbers of characters, emoticons usage and capital letters) as well as Part-Of-Speech analysis. Starting from n-grams or Bag-of-Words (BoW) approaches, a few number of participants extracted topic words and proposed to use automatic readability indices (Coleman-Liau, Rix Readability index, Gunning Fox index, Flesch-Kinkaid). Recently, Werren et al [24] proposed an important number of experiments combining several features (Flesch-Kinkaid readilibity index), psycholinguistic concepts (using MRC and LIWC [25] features) and distance metrics (Cosine, OkapiBM25) evaluated on the PAN 2014 dataset. Participants also considered these readability indexes in linear classifier such as SVM and libLINEAR. PAN participants approached the task using ML techniques. All these features were used to feed several algorithms such as logistic regression, logic boost, multinominal Naïve Bayes, etc.

The RepLab 2014 *"Author Ranking"* task was specifically focused on influence [19], as explained in more details in Section III. Participants mainly considered the tweet textual content to model each user, and applied various ML tools. They used Logistic Regression, Logic Boost, Random and Rotation Forests, Multi-layer Perceptron and Linear approaches such libLINEAR and SVM, over a large variety of features. The UTDBRG group obtained the best performance by using Trending Topics Information, assuming that *Influencers* tweet mainly about *"Hot Topics"*. Based on the assumption that *Influencers* tend to use specific terms in their tweets, the LIA group [26] opted to model each user based on the textual content associated to his tweets. Using k-Nearest Neighbors (k-NN), they then matched each user to the most similar ones

in the training set. The LyS group [27] used specific (such as URLs, verified account tag, user image) and quantitative (number of followers) profile meta-data. See Table IV for the numerical results. Moreover, UAMCLYR also considered NLP *Quantitative Stylistic* and *Behavioral* features extracted from tweet contents and extended their approach after the challenge [28].

III. REPLAB CHALLENGE

The CLEF RepLab 2014 dataset [19] was designed for an influence ranking challenge organized in the context of the Conference and Labs of the Evaluation Forum[2] (CLEF). As mentioned before, we use these data for our own experiments. In this Section, we first describe the context of the challenge and the data, then how the performance were evaluated, and we also discuss the obtained results. Finally, we present a classification variant of the task.

A. Data and task

The RepLab dataset contains users manually labeled by specialists from Llorente & Cuenca[3], a leading Spanish e-Reputation firm. These users were annotated according to their perceived real-world (offline) influence, and not by considering specifically their Twitter account. The annotation is binary: a user is either *Influencer* or *Not-Influencer*. The dataset contains a *training set* of 2500 users, including 796 *Influencers*, and a *testing set* of 4500 users, including 1563 *Influencers*. It also contains the 600 last tweets of each user at the crawling time.

Given the low number of real *Influencers*, the RepLab organizers modeled the issue as a search problem restrained to the *Automotive* and *Banking* domains. In other words, the dataset was split in two, depending on the main activity domain of the considered users. The objective was to rank the users in both domain in the decreasing order of influence. Both domains are balanced, with 2323 (testing) and 1186 (training) users for the Automotive domain, and 2482 (testing) and 1314 (training) for the Banking domain.

The organizers proposed a baseline consisting in ranking the users by descending number of followers. Basically, this amounts to considering that the more a given user has followers, the more he is expected to be influential.

B. Evaluation

The RepLab framework [19] uses the *Mean Average Precision* (MAP) to evaluate the estimated rankings. MAP allows comparing an ordered vector (output of a submitted method) to a binary reference (manually annotated data). The MAP is computed as follows:

$$MAP = \frac{1}{n} \sum_{i=1}^{N} p(i) R(i) \tag{1}$$

where N is the total number of users, n the number of influencers correctly found (i.e. true positives), $p(i)$ the precision

at rank i (i.e. when considering the first i users found) and $R(i)$ is 1 if the ith user is influential, and 0 otherwise.

The MAP is computed separately for each domain, and RepLab participants were compared according to the Average MAP processed over both domains.

C. Results

According to the official evaluation, the proposal from UT-DBRG obtained the highest MAP for the Automotive domain (.721) and the best Average MAP among all participants (.565). The proposal from LIA obtained the highest MAP for the Banking domain (.446). The performance differences observed between domains are likely due to the fact one domain is more difficult to process than the other one. The *Followers baseline* remains lower than most of submitted systems, achieving a MAP of .370 for Automotive and .385 for Banking. All these values are summarized in Table IV, in order to compare them with our own results.

D. Classification Variant

Because the reference itself is only binary, the RepLab ordering task can alternatively be seen as a binary classification problem, consisting in deciding if a user is an *Influencer* or not. However, this was not a part of the original challenge. Ramirez *et al*. [28] recently proposed a method to tackle this issue. We will also consider this variant of the problem in the present article.

To evaluate the classifier performance, Ramirez *et al*. used the F-Score averaged over both classes, based on the Precision and Recall processed for each class, which is typical in categorization tasks. The *Macro Averaged F-Score* is calculated as follows:

$$F = \frac{1}{k} \sum_c \frac{2(P_c \times R_c)}{P_c + R_c} \tag{2}$$

where P_c and R_c are the Precision and Recall obtained for class c, respectively, and k is the number of classes (for us: 2). The performance is considered for each domain (Banking and Automotive), as well as averaged over both domains. It gives an overview of the system ability to recover information from each class.

Ramirez *et al*. do not use any baseline to assess their results. Nevertheless, the imbalance between the influencer (31%) and non-influencer (69%) in the dataset leads to a strong non-informative baseline which simply consists in putting all users in the majority class (non-influencers).

IV. FEATURES

For our experiments, we selected the most widespread features found in SNA and NLP works related to the characterization of Twitter profiles. In this Section, we describe them, before defining our own new features.

A. Traditional Features

As shown in Table I, we investigated a large selection of traditional features taken from both SNA and NLP works. We gathered these features in several categories, all describing specific aspects of a Twitter account. Features 1–3 describe how active a user is (number of tweets posted...). Features 4–8 are related to the way a user is connected with to the rest of the Twitter network (number of friends...). Features 9–14 measure how the user takes advantage of Twitter-specific linking methods (number of mentions, URL...). Features 15–23 are related to the tweets themselves (their size, frequency...). Features 24–28 directly represent the fields composing a user profile (presence of a picture, personal Website...). Finally, feature 29 describes the tweets content from a purely NLP perspective.

All feature names are self-explanatory, except for the last one. We defined our term-weighting using the classic *Frequency-Inverse Document Frequency* (TF-IDF) [29], combined with the *Gini purity criterion* [30]. The purity G_i of a word i is defined as follows:

$$G_i = \sum_{c \in \mathbb{C}} \mathbb{P}^2(i|c) = \sum_{c \in \mathbb{C}} \left(\frac{DF_c(i)}{DF(i)} \right)^2 \qquad (3)$$

where $\mathbb{C}$ is the set of classes, $DF(i)$ is the number of documents (in the training set) containing the word i, and $DF_c(i)$ is the number of documents (in the training set) annotated with class c and containing word i. The Gini criterion is used to weight the contribution $\omega_{i,d}$ of each term i in document d:

$$\omega_{i,d} = TF_{i,d} \times log(\frac{N}{DF(i)}) \times G_i \qquad (4)$$

as well as the contribution $\omega_{i,c}$ of each term i in class c:

$$\omega_{i,c} = DF_{i,c} \times log(\frac{N}{DF(i)}) \times G_i \qquad (5)$$

where N is the number of documents in the training set. Both weights were used in two different ways, as described in Section V-A.

B. Original Features

We also used some additional features, which seemed relevant in the context of influence prediction. Those are presented in Table II. Features 30–31 are based on data retrieved out of Twitter: the Klout score and the number of Google results pointing at the user's personal website.

The rest of the features are related to the notion of user word cooccurrence matrix. In NLP, word occurrence frequency is widely used to characterize texts or groups of texts. The idea here is to proceed similarly, but with word *cooccurrences*, and to use this approach to describe the users. Put differently, for each user, we process a matrix representing how many times each word pair appears consecutively over all the tweets he posted. Each unique tweet content is lower-cased and cleaned by removing hypertext links, stop-words and punctuation marks. We ignored words with 1 or 2 letters.

TABLE I
SELECTION OF TRADITIONAL FEATURES.

Category	Features
User activity	Numbers of: 1) Tweets (or statuses); 2) Lists containing the user; 3) Tweets marked as favorites.
Local topology	4) Size of the friends set; 5) Size of the followers set; 6) Size of the intersection of the $5,000$ most recent friends and followers sets; 7) Standard deviation of the $5,000$ most recent friends' identifiers; 8) Standard deviation of the $5,000$ most recent followers' identifiers.
Stylistic aspects	Average numbers of: 9) Hashtags per tweet; 10) URLs per tweet; 11) Mentions per tweet; 12) Distinct hashtags per tweet; 13) Distinct URLs per tweet; 14) Distinct users mentioned per tweet.
Tweets characteristics	15) Average and standard deviation of the number of characters per tweets; 16) Minimum, maximum, average and standard deviation of the number of retweets; 17) Minimum, maximum, average and standard deviation of the number of favorites; 18) Proportion of retweets among tweets; 19) Average and standard deviation of the time gap between tweets, in seconds; 20) Proportion of geolocated tweets; 21) Number of distinct geolocations used; 22) Proportion of tweets that are replies; 23) Number of distinct users to whom the user replied.
Profile fields	24) Picture in the profile (Boolean); 25) Verified Account (Boolean); 26) Allow Contribution (Boolean); 27) URL in the profile (Boolean); 28) Description size;
Occurrence-based term weighting	29) TF×IDF×Gini weights.

Feature 42 corresponds to the Euclidean distance between all pairs of matrices, i.e. all pairs of users. Moreover, using each of these matrices as an adjacency matrix, we additionally built a collection of graphs called cooccurrence networks. We described each graph through a set of classic nodal topological measures, represented in Table II as features 32–41. During our experiments, we used these measures under two forms: a vector of values, each one describing one node in the considered graph; and their arithmetic mean.

The selected measures are complementary, certain are based on the *local* topology (degree, transitivity), some are *global* (betweenness, closeness, Eigenvector and subgraph centralities, eccentricity), and the others rely on the network community structure, and are therefore defined at an *intermediary* level (embeddedness, within-module degree, participation

Category	Features
External data	30) Klout Score; 31) Number of Google results pointing at the user's personal Website.
Cooccurrence-based term weighting	Individual and average values of: 32) Degree; 33) Betweenness centrality; 34) Closeness centrality; 35) Eigenvector centrality; 36) Subgraph centrality; 37) Eccentricity; 38) Transitivity; 39) Embeddedness; 40) Within Module Degree; 41) Participation Coefficient.
Cooccurence matrix distance	42) Euclidean distance between matrices.

coefficient). In their description, we note $G = (V, E)$ the considered cooccurrence graph, where V and E are its sets of nodes and links, respectively.

The *Degree* measure $d(u)$ is quite straightforward: it is the number of links attached to a node u. So in our case, it can be interpreted as the number of words co-occurring with the word of interest. More formally, we note $N(u) = \{v \in V : \{u, v\} \in E\}$ the *neighborhood* of node u, i.e. the set of nodes connected to u in G. The degree $d(u) = |N(u)|$ of a node u is the cardinality of its neighborhood, i.e. its number of neighbors.

The *Betweenness* centrality $C_b(u)$ is a measure of accessibility [31]. It amounts to the number of shortest paths going through u to connect other nodes: $C_b(u) = \sum_{v<w} \sigma_{vw}(u)/\sigma_{vw}$, where σ_{vw} is the total number of shortest paths from node v to node w, and $\sigma_{vw}(u)$ is the number of shortest paths from v to w running through node u.

The *Closeness* centrality $C_c(u)$ quantifies how near a node u is to the rest of the network [32]: $C_c(u) = 1/\sum_{v \in V} dist(u, v)$, where $dist(u, v)$ is the *geodesic distance* between nodes u and v, i.e. the length of the shortest path between these nodes.

The *Eigenvector* centrality $C_e(u)$ measures the influence of a node u in the network based on the spectrum of its adjacency matrix. The Eigenvector centrality of each node is proportional to the sum of the centrality of its neighbors [33]:

$$C_e(u) = \frac{1}{\lambda} \sum_{v \in N(u)} C_e(v) \qquad (6)$$

Here, λ is the largest Eigenvalue of the graph adjacency matrix.

The *Subgraph* centrality $C_s(u)$ is based on the number of closed walks containing a node u [34]. Closed walks are used here as proxies to represent subgraphs (both cyclic and acyclic) of a certain size. When computing the centrality, each walk is given a weight which gets exponentially smaller as a function of its length.

$$C_s(u) = \sum_{\ell=0}^{\infty} \frac{\left(A^\ell\right)_{uu}}{\ell!} \qquad (7)$$

Where A is the adjacency matrix of G, and therefore $\left(A^\ell\right)_{uu}$ corresponds to the number of closed walks containing u.

The *Eccentricity* $E(u)$ of a node u is its furthest (geodesic) distance to any other node in the network [35].

The *Local Transitivity* $T(u)$ of a node u is obtained by dividing the number of links existing among its neighbors, by the maximal number of links that could exist if all of them were connected [36]:

$$T(u) = \frac{|\{\{v, w\} \in E : v \in N(u) \wedge w \in N(u)\}|}{d(u)(d(u) - 1)/2} \qquad (8)$$

Where the denominator corresponds to the binomial coefficient $\binom{d(u)}{2}$. This measure ranges from 0 (no connected neighbors) to 1 (all neighbors are connected).

The *Embeddedness* $e(u)$ represents the proportion of neighbors of a node u belonging to its own community [37]. The community structure of a network corresponds to a partition of its node set, defined in such a way that a maximum of links are located *inside* the parts while a minimum of them lie *between* the parts. We note $c(u)$ the community of node u, i.e. the parts that contains u. Based on this, we can define the *internal neighborhood* of a node u as the subset of its neighborhood located in its own community: $N^{int}(u) = N(u) \cap c(u)$. Then, $d^{int}(u) = |N^{int}(u)|$ is the *internal degree*. Finally, the embeddedness is the ratio $e(v) = d_{int}(v)/d(v)$. It ranges from 0 (no neighbors in the node community) to 1 (all neighbors in the node community).

The two last measures were proposed by Guimerà & Amaral [38] to characterize the community role of nodes. For a node u, the *Within Module Degree* $z(u)$ is defined as the z-score of the internal degree, processed relatively to its community $c(u)$:

$$z(u) = \frac{d_{int}(u) - \mu(d_{int}, c(u))}{\sigma(d_{int}, c(u))} \qquad (9)$$

Where μ and σ denote the mean and standard deviation of d_{int} over all nodes belonging to the community of u, respectively. This measure expresses how much a node is connected to other nodes in its community, relatively to this community.

The *Participation Coefficient* is based on the notion of community degree : $d_i(u) = |N(u) \cap C_i|$. This corresponds to the number of links a node u has with nodes in C_i, namely nodes belonging to community number i. The participation coefficient is defined as:

$$P(u) = 1 - \sum_{1 \leq i \leq k} \left(\frac{d_i(u)}{d(u)}\right)^2 \qquad (10)$$

Where k is the number of communities. P characterizes the distribution of the neighbors of a node over the community

structure. To detect communities, we applied the InfoMap algorithm [39], which was deemed very efficient in previous studies [40].

V. RESULTS AND DISCUSSIONS

In this Section, we present the results we obtained on the RepLab dataset. The analysis tools we applied are relatively standard, so we quickly describe them first. Afterwards, we consider the results obtained for the classification task, then the ranking one.

A. Experimental Setup

The large variety of features we considered required us to process them in different ways. Most of them are scalars, in the sense each user is represented by a single numerical value (Features 1–28, 30–31 and averaged Features 32–41). A few features are vectors, i.e. each user is represented by a series of values (Features 29 and 32–41). Finally, Feature 42 is particular, since it is actually constituted by the distances between all pairs of users.

First, in order to figure out whether or not the scalar features were relevant, we ran a Principal Component Analysis (PCA). Its first three components explain a bit less than 50% of the variance. The first plane, displayed in Figure 1, shows these features cannot be used to discriminate linearly *Influencers* from other users (the other components confirm this).

We thus turn to non-linear classifiers under the form of kernelized SVMs (RBF, Polynomial and Sigmoid kernels). We used the logistic regression trained with each scalar feature alone, as well as with all combinations of scalar features within each category defined by us (as described in Tables I and II), and with all scalar features at once. Furthermore, we considered separately users from the two domains considered in the dataset: *Banking* and *Automative*.

Regarding Feature 29 (terms weighting) we investigated two user-profile definitions: *User-as-document* [41] (noted UaD in the rest of the article) and *Bag-of-tweets* (BoT). With the UaD approach, all tweets from each user belonging to a given class are merged to create one large document. Users are then compared by computing the similarity between both corresponding documents. Ranking is obtained using the probabilities associated to the class *Influencer*. The BoT approach consists in considering a binary classification problem for each tweet. The Bag-of-Words representation is used for each individual tweet, as well as for each class in each domain. For instance, the *Influential* Banking class BoW is built upon all tweets posted by influential users in the training set for the Banking domain. We compute the similarity between each tweet BoW and each class BoW. Then, a user is deemed influential if a majority of his tweets are themselves considered influential. Ranking is achieved by counting the number of tweets classified as influential for the considered user.

We computed document-to-class similarities using Cosine distance as follows:

$$cos(d, c) = \frac{\sum\limits_{i \in d \cap c} \omega_{i,d} \times \omega_{i,c}}{\sqrt{\sum\limits_{i \in d} \omega_{i,d}^2 \times \sum\limits_{i \in c} \omega_{i,c}^2}} \qquad (11)$$

where d and c are the considered document and class, respectively, and the ω are those defined in Section IV.

Because it is distance-based, Feature 42 had to be processed separately. We used a k-NN based classification consisting in matching each profile of the test collection to the k closest profiles of the training set. As mentioned before, the profiles were compared based on the Euclidean distance computed between the corresponding word cooccurrence matrices. We tried different values of k, ranging from 1 to 20.

B. Classification

The kernelized SVMs we applied did not converge when considering scalar features: individually, by category, by combining categories and all together. We obtained the same behavior for vector Features 32–41. This means those tools could not find any non-linear separation of our two classes using this information. Those results were confirmed by the logistic regressions. Indeed, none of the trained classifiers performed better than the most-frequent class baseline (all user as non-influential). Random forests gave the same results. Meanwhile, as stated in II, these classifiers usually perform very well for this type of task.

However, we obtained some results for the remaining features, as displayed in Table III. The classification performances are shown in terms of F-Score for each domain and averaged over both domains. For comparison purposes, we also report the best results obtained by [28] using SVM, and by the *LIA* group based on tweets content (Section III).

The NLP cosine-based approach applied to Feature 29 shows competitive performances, noticeably higher than the baselines. The BoT approach obtained state-of-the-art results while the UaD method outperformed all performances reported for this task, up to our knowledge. As mentioned before,
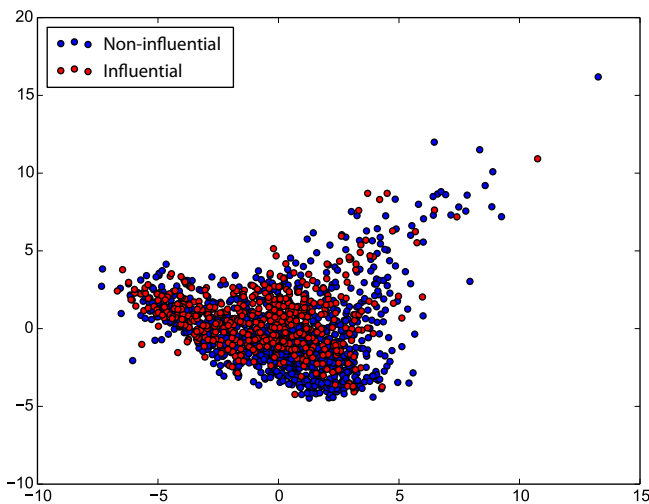

Fig. 1. Principal component analysis, first factorial plan.

TABLE III
CLASSIFICATION PERFORMANCES ORDERED BY AVERAGE F-SCORE.

Method	Automotive	Banking	Average
Feature 29 UaD	.833	.751	.792
LIA	.702	.726	.714
Ramirez *et al.*	.696	.693	.694
Feature 29 BoT	.725	.641	.683
MF-Baseline	.500	.500	.500
Feature 42	.403	.417	.410

TABLE IV
RANKING PERFORMANCES ORDERED BY AVERAGE MAP.

#Method	Automotive	Banking	Average
Feature 29 UaD	.803	.626	.714
Feature 29 BoT	.626	.504	.565
UTDBRG	.721	.410	.565
LIA	.502	.446	.476
Feature 1	.332	.449	.385
Best Regression	.424	.338	.381
RepLab Baseline	.370	.385	.378
Feature 42	.298	.300	.299
Feature 30	.304	.275	.289

Feature 42 was processed by the k-NN method. The different k values we tested did not lead to significantly different results. The other tested features were not able to reach the performance level defined as a baseline, and thus neither those obtained by state-of-the-art work.

C. Ranking

The results obtained for the ranking task are displayed in Table IV in terms of MAP, for each domain and averaged over both domains. The *UTDBRG* row corresponds to the scores obtained at RepLab by the UTDBRG group, which reached the highest global performance and the best MAP for Automotive. This high performance for the Automotive domain with the trending topics information probably reflects a tendency for Influencers to be up-to-date with the latest news relative to brand products and innovations. This statement is not valid for Banks, where we can suppose that influence is based on more specialized and technical discussions. This is potentially why the *LIA* approach based on tweets content obtained a good result for this domain, as mentioned in Section III.

Regarding our data, we evaluated the logistic regression trained with each scalar feature alone, with each one of their categories, with each combination of category, and with all scalar features at once. The best results are represented on the row *Best Regression*, and were obtained by combining features of the following categories: user activity, profile fields, stylistic aspects (Table I) and external data (Table II).

For each numerical scalar feature, we also considered the features values as a ranking method. The best results were obtained using the number of tweets posted by each user (Feature 1). Although its average MAP is just above the baseline, the performance obtained for the Banking domain is above the best state-of-the-art results. Thus, we may consider this feature as the new baseline of this specific domain. All others similarly processed features remain lower than the official baseline. The results obtained for Feature 30 reflect very poor rankings. This is very surprising, because this feature is the Klout Score, which was precisely designed to measure influence.

The results obtained for Feature 42 (cooccurrence matrices) is slightly better than for the Klout Score. Like before, Feature 42 was processed by the k-NN approach. Again, the various tested k values did not lead to significantly different results. The performance presented in Table IV is the best we obtained.

The cosine-based approach applied to Feature 29 led to very interesting results. The BoT method obtained an average state-of-the-art performance, while the UaD method reaches very

high average MAP values, even larger than the state-of-the-art, be it domain-wise (for both Automotive and Banking) or in average. This means describing a user in function of the vocabulary he uses over all his tweets retains the information necessary to decide how influential he is. In other words, influencers are characterized by a certain editorial behavior.

From these results, we claim that typical SNA features classically used to detect spammers, social capitalists or influential Twitter users, are not very relevant to detect real-life influencers based on Twitter data. In other terms, they may only characterize influence perceived on Twitter. The results were much better with the NLP approach consisting in representing a user under various forms of bags-of-words. In particular, our User-as-a-document approach gives far better results than the best state-of-the-art approaches. Put differently, the way a user writes his tweets may be related to his offline influence, at least for the studied domains. However, our attempt to extend this occurrence-based approach to a cooccurrence-based one using graph measures did not lead to good performances.

VI. CONCLUSION

In this article, we have investigated a wide range of methods and features to tackle the tasks of identifying real-life (offline) Influencers and ranking people according to their influence based on Twitter-related data. We can highlight three main results. First, we showed that classical SNA features used to detect spammers, social capitalists or users influential *on Twitter* do not give any significant results. They are able to predict influence considered internally to Twitter itself, but not in real-life. Still, the number of tweets posted by a user seems to constitute a new, better baseline in the banking domain according to our study. Our second result is to have shown that, like the previously mentioned SNA features, the Klout Score does not allow to predict real-life influence neither.

Third, we proposed an NLP approach consisting in representing a user under various forms of bags-of-words, which led to much better performances. In particular, our User-as-a-document method reaches much higher MAP values than the best state-of-the-art approaches. From this result, we can suppose the way a user writes his tweets is related to his real-life influence, at least for the studied domains. This would confirm assumptions previously expressed in the literature regarding the fact users from specific domains behave and write in their own specific way.

Our work can be criticized in several ways, though. We used a wide range of features, but it is still not exhaustive. We plan to complete this in our next work. Also on the feature aspect, because of the good results obtained using word occurrence-based features, we tried to take advantage of word cooccurrences. However, this did not result in good performances. But this path can still be further explored, by using other graph measures, or different methods to build the cooccurrence matrix, for instance by considering higher order word neighborhoods, or even word triplets.

Moreover, our results are valid only for the considered dataset. This means they are restricted to the domains it describes (Automotive and Banking), and also they are only as good as the manual annotation of the data. Actually, in RepLab 2014 [19], the organizers were not able to conclude on significant differences between certain participants due to the number of considered domains. This point should be solved quickly though, through the 2015 edition of PAN[4].

References

[1] A. Java, X. Song, T. Finin, and B. Tseng, "Why we twitter: understanding microblogging usage and communities," in *WebKDD/SNA-KDD*, 2007, pp. 56–65.

[2] D. Boyd, S. Golder, and G. Lotan, "Tweet, tweet, retweet: Conversational aspects of retweeting on twitter," in *HICSS*, 2010, pp. 1–10.

[3] S. A. Myers, A. Sharma, P. Gupta, and J. Lin, "Information network or social network?: The structure of the twitter follow graph," in *WWW companion*, 2014, pp. 493–498.

[4] Moderateur, "Figures about twitter in 2015," 01 2015, http://www.blogdumoderateur.com/chiffres-twitter/.

[5] S. Ghosh, B. Viswanath, F. Kooti, N. Sharma, G. Korlam, F. Benevenuto, N. Ganguly, and K. Gummadi, "Understanding and combating link farming in the Twitter social network," in *WWW*, 2012, pp. 61–70.

[6] C. Hendricks and D. Schill, *Presidential Campaigning and Social Media: An Analysis of the 2012 Campaign*. Oxford University Press, 2014.

[7] S. Burton and A. Soboleva, "Interactive or reactive? : marketing with twitter," *J. Consum. Mark.*, vol. 28, no. 7, pp. 491–499, 2011.

[8] Klout, "Klout, the standard for influence," http://www.klout.com.

[9] Kred, "Kred story," http://www.kred.com.

[10] M. Danisch, N. Dugué, and A. Perez, "On the importance of considering social capitalism when measuring influence on twitter," in *Behavioral, Economic, and Socio-Cultural Computing*, 2014.

[11] J. Messias, L. Schmidt, R. Oliveira, and F. Benevenuto, "You followed my bot! transforming robots into influential users in Twitter," *First Monday*, vol. 18, no. 7, 2013.

[12] K. Lee, P. Tamilarasan, and J. Caverlee, "Crowdturfers, campaigns, and social media: Tracking and revealing crowdsourced manipulation of social media." in *ICWSM*, 2013.

[13] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on twitter," in *WSDM*, 2011, pp. 65–74.

[14] K. Lee, J. Mahmud, J. Chen, M. Zhou, and J. Nichols, "Who will retweet this," in *19th IUI*, 2014, pp. 247–256.

[15] S. Pramanik, M. Danisch, Q. Wang, and B. Mitra, "An empirical approach towards an efficient "whom to mention?" twitter app," *Twitter for Research, 1st International Interdisciplinary Conference*, 2015.

[16] B. Suh, L. Hong, P. Pirolli, and E. H. Chi, "Want to be retweeted? large scale analytics on factors impacting retweet in twitter network," in *Social Computing*, 2010, pp. 177–184.

[17] R. M. Bond, C. J. Fariss, J. J. Jones, A. D. Kramer, C. Marlow, J. E. Settle, and J. H. Fowler, "A 61-million-person experiment in social influence and political mobilization," *Nature*, vol. 489, no. 7415, pp. 295–298, 2012.

[18] F. Rangel, P. Rosso, I. Chugur, M. Potthast, M. Trenkmann, B. Stein, B. Verhoeven, and W. Daelemans, "Overview of the 2nd author profiling task at pan 2014," in *CLEF Evaluation Labs and Workshop*, 2014.

[19] E. Amigó, J. Carrillo-de Albornoz, I. Chugur, A. Corujo, J. Gonzalo, E. Meij, M. de Rijke, and D. Spina, "Overview of replab 2014: author profiling and reputation dimensions for online reputation management," in *Information Access Evaluation. Multilinguality, Multimodality, and Interaction*. Springer, 2014, pp. 307–322.

[20] K. Lee, B. D. Eoff, and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on twitter." in *ICWSM*, 2011.

[21] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi, "Measuring user influence in Twitter: The million follower fallacy," in *ICWSM*, 2010.

[22] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *WSDM*, 2010, pp. 261–270.

[23] I. Anger and C. Kittl, "Measuring influence on Twitter," in *11th International Conference on Knowledge Management and Knowledge Technologies*, 2011, pp. 1–4.

[24] E. R. Weren, A. U. Kauer, L. Mizusaki, V. P. Moreira, J. P. M. de Oliveira, and L. K. Wives, "Examining multiple features for author profiling," *J. Inf. Data Manage.*, vol. 5, no. 3, p. 266, 2014.

[25] A. Tumasjan, T. Sprenger, P. Sandner, and I. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment," in *ICWSM*, 2010, pp. 178–185.

[26] J.-V. Cossu, K. Janod, E. Ferreira, J. Gaillard, and M. El-Bèze, "Lia@replab 2014: 10 methods for 3 tasks," *4th International Conference of the CLEF initiative*, 2014.

[27] D. Vilares, M. Hermo, M. A. Alonso, C. Gómez-Rodrıguez, and J. Vilares, "Lys at clef replab 2014: Creating the state of the art in author influence ranking and reputation classification on twitter," in *4th International Conference of the CLEF initiative*, 2014, pp. 1468–1478.

[28] G. Ramírez-de-la Rosa, E. Villatoro-Tello, H. Jiménez-Salazar, and C. Sánchez-Sánchez, "Towards automatic detection of user influence in twitter by means of stylistic and behavioral features," in *Human-Inspired Computing and Its Applications*. Springer, 2014, pp. 245–256.

[29] K. Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval," *J. Doc.*, vol. 28, no. 1, pp. 11–21, 1972.

[30] E. Gaussier and F. Yvon, "Opinion detection as a topic classification problem," in *Textual Information Access: Statistical Models*. John Wiley & Son, 2013, ch. 9, pp. 245–256.

[31] L. C. Freeman, D. Roeder, and R. R. Mulholland, "Centrality in social networks ii: Experimental results," *Social Networks*, vol. 2, no. 2, pp. 119–141, 1979.

[32] A. Bavelas, "Communication patterns in task-oriented groups," *Journal of the Acoustical Society of America*, vol. 22, no. 6, pp. 725–730, 1950.

[33] P. F. Bonacich, "Power and centrality: A family of measures," *American Journal of Sociology*, vol. 92, pp. 1170–1182, 1987.

[34] E. Estrada and J. A. Rodriguez-Velazquez, "Subgraph centrality in complex networks," *Physical Review E*, vol. 71, no. 5, p. 056103, 2005.

[35] F. Harary, *Graph Theory*. Addison-Wesley, 1969.

[36] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[37] A. Lancichinetti, M. Kivelä, J. Saramäki, and S. Fortunato, "Characterizing the community structure of complex networks," *PLoS ONE*, vol. 5, no. 8, p. e11976, 2010.

[38] R. Guimerà and L. N. Amaral, "Cartography of complex networks: modules and universal roles," *J. Stat. Mech.*, vol. 02, p. P02001, 2005.

[39] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, p. 1118, 2008.

[40] G. K. Orman, V. Labatut, and H. Cherifi, "Comparative evaluation of community detection algorithms: A topological approach," *Journal of Statistical Mechanics*, vol. 8, p. P08001, 2012.

[41] Y.-M. Kim, J. Velcin, S. Bonnevay, and M.-A. Rizoiu, "Temporal multinomial mixture for instance-oriented evolutionary clustering," in *Advances in Information Retrieval*, 2015.

---

[4]http://pan.webis.de/