NLP-based classifiers to generalize expert assessments in E-Reputation

Jean-Valère Cossu[1], Emmanuel Ferreira[1], Killian Janod[1,2], Julien Gaillard[1] and Marc El-Bèze[1]

LIA/Université d'Avignon et des Pays de Vaucluse[1]
39 chemin des Meinajaries, Agroparc BP 91228, 84911 Avignon cedex 9, France
ORKIS - Aix en Provence (France)[2]
firstname.name@univ-avignon.fr[1]
kjanod@orkis.com[2]

Abstract. *Online Reputation Management*(ORM) is currently dominated by expert abilities. One of the great challenges is to effectively collect annotated training samples, especially to be able to generalize a small pool of expert feedback from area scale to a more global scale. One possible solution is to use advanced *Machine Learning* (ML) techniques, to select annotations from training samples, and propagate effectively and concisely. We focus on the critical issue of understanding the different levels of annotations. Using the framework proposed by the RepLab contest we present a considerable number of experiments in Reputation Monitoring and Author Profiling. The proposed methods rely on a large variety of *Natural Language Processing* (NLP) methods exploiting tweet contents and some background contextual information. We show that simple algorithms only considering tweets content are effective against state-of-the-art techniques.

1 Introduction

Analyzing a company's and an individual's reputation is a difficult end-user oriented problem, requiring complex modeling. Experts involved in this modeling might generate features that computers are not able to differentiate or capture. ORM has become a key component for an entity's communication strategy with the growing influence of information available on social networks [1]. Reputation managers still have to monitor and analyze social data related to their brand for alarm signals manually and take immediate action to avoid damages on the reputation of their clients on key issues. Hybrid approaches have been proposed in works such as [2, **?**] for automatic annotation under expert supervision and estimation of the gain in using support tools. However, as the field is relatively new, the algorithmic support for ORM is still limited.

Last RepLab[1] [3, 4] and TASS[2] [5] evaluations have shown significant algorithmic advances in several aspects related to ORM tasks such as filtering,

[1] http://www.limosine-project.eu/events/replab2013
[2] http://www.daedalus.es/TASS2013/about.php

polarity for reputation (*Sentiment Analysis*) and clustering (the so-called '*Topic Detection*' by the organizers of RepLab). It became quite clear that systems have achieved high classification results ([6–8]) that may not reflect annotators' assessments variety. Nevertheless, other aspects such as *polarity analysis* at politics-entity level, *tweets ranking*, *dimensions detection*, *socioeconomic classification* for *Author Profiling* (with the PAN contest [9]) still require further progress. This is mainly because these aspects are vague, subjective and may depend on each expert. Then, it becomes harder to automatically predict an exact class when the models are difficult to understand and subject to diversity.

We proposed experiments for all the tasks in the 2013 and 2014 editions of RepLab. In this paper, we deal with *Reputation Alert Detection*, *Reputation Dimension Assignment* and *Author Profiling* (AP). We will not focus on the remaining tasks since the issues have been considered as partially solved [6–8]. Our main objective is to extract sets of textual contents requiring a particular attention from a reputation manager. By doing so, we aim at guiding reputation experts to understand why a decision should be taken after these tweets. For this purpose we also need to determine the importance of an author and its type with regard to its *'bag-of-tweets'*. We use *NLP-based classifiers* to project each tweet in a *multidimensional reputation* space to generalize the expert's point-of-view. Then, we determine whether this expertise concerns the topic of a particular message or the Influence of an author.

The rest of this paper is organized as follows: Section 2 gives an overview of related work and establishes further motivation for our work. In Sections 3 - 4, we provide details of our approaches to tackle *E-Reputation tasks* and *Author Profiling*. A discussion of our results is provided in Section 5. Finally, Section 6 give our conclusions on our work and open several perspectives.

2 Related work

To our knowledge, most of the contributions on ORM were proposed in the last editions of RepLab [3, 4] and TASS [5]. Others contributions took place in the context of major national elections in Mexico [10], France and Spain respectively on the behalf of the Imagiweb project [11, 12] and TASS campaign [13].

RepLab'2014's reputation dimensions classification task is a complement to the *Topic Detection* of the previous edition. It also comparable to the *Target-Oriented Opinion* defined in the Imagiweb [11, 12] project as it is nearer to a stress classification of the aspects of the entity under public scrutiny. Where these stresses are defined by experts (stakeholders, reputation managers, communication adviser or scientists) and only reflect their own interest, which may differ from the real information carried in the tweets (topic, event...). All these works mainly rely on supervised classification methods based on tweet content and its more or less complex pre-processing [6, 14]. Moreover, ORM issues are tackled at a different granularity level: global approaches, domain or entity specific models. There are also three sub-levels of approach: the use of meta-data, the human involvement in the systems (so-called hybrid categorization approaches) and the

use of additional lexical or linguistic resources. In this light, *SibTex* [15] investigated both domain and language specific approaches. They reported a slight improvement in using domain or language dedicated knowledge bases in their *k-NN* approach. This statement may confirm that any information deemed important should be considered to feed systems (in some way it can be considered as a form of enrichment) rather than focusing on a single entity. It then requires a huge amount of annotated data to provide stable hypotheses.

PAN [9] provides a nice overview of AP recent progress. More generally in RepLab *Social Network Analysis* techniques were used for both tweets and user ranking [16, 14, 17]. Assuming that *Influencers* tweet mainly about *'Hot Topics' UTDBRG* group obtained the best performance by using *Trending Topics Information*. Some works investigated extended tweet-representation to consider information beyond the tweet textual content such as pseudo-relevant term expansion [18] as well as Wikipedia-concept term expansion [19] to enrich the tweets and improve a *Random Forest classifier*. These works imply a heavy involvement to induce rules or to build the resources used. We also experimented a joint work linking *Tweet Clustering* [20] and *Dimensions Classification* [21] to *Priority Detection* over a *NLP-based Classification*.

Nevertheless, according to RepLab organizers [3, 4], it remains difficult to find a correspondence between performances and algorithm or features used. Moreover, the amount of research dedicated to understanding the experts' stress effects on mis-classified tweets is very limited.

3 Reputation Monitoring

The RepLab [3, 4] framework propose a complete Reputation Monitoring challenge for more than 61 entities drawn from four domains: Automotive, Banking, Music and University. We approach Reputation Monitoring as the following cascade: for a given set of tweets in a certain time span, we have to identify opinions in key topics whatever the entity concerned as it could be done manually by reputation management experts just in reading the tweets stream. Then systems have to identify tweet clusters (each cluster represents a topic/event/issue/conversation). As these aspects are not know *a-priori* we are far from a typical categorization task or standard Topic Detection problem. The clusters are then ranked into Priority level (*Alert, Important, Unimportant*). Additionally systems have to look for positive or negative implications of the contents on the entity's reputation and finally tweets are categorized according to their *Reputation Dimensions* using standards given by the Reputation Institute's Reptrak framework [3]. From the perspective of reputation management, *'Reputation Alerts'* which have immediate and negative effects on the entity's reputation must be clearly identified and detected early enough to prevent the number of tweets growth over these topics.

[3] `http://www.reputationinstitute.com/about-reputation-institute/the-reptrak-framework`

In this paper, we intend to extend the work done in the RepLab context. We aim to observe the effect of tweet content and its engineering on ORM and ML methods. Our main contribution is the assessment of tweet content efficiency for E-Reputation Analysis.

3.1 Approaches

A short description and preliminary results obtained in each task with our approaches have been presented in [22, 23]. Class hypotheses are generated by the following systems:

- Cosine is considered as a more or less lightweight statistical baseline;
- SVM is used as a state-of-the-art classification baseline;
- We also propose a CRF-based approach to extend the bag-of-words.

Within those systems we use a Word2Vec models as generalization engine.

Terms Weighting. The features used by our baselines proposals are words, bi-grams and tri-grams. They compose the tweet discriminant bag-of-words representation. We use Term Frequency-Inverse Document Frequency (TF-IDF) [24] combined with the Gini purity criteria, as several works reported improvements using this association [25]. Purity of a word G_i is defined with the Gini criterion as follows (1):

$$G_i = \sum_{c \in \mathbb{C}} \mathbb{P}^2(i|c) = \sum_{c \in \mathbb{C}} \left(\frac{DF_i(c)}{DF_(i)} \right)^2 \tag{1}$$

where C is the set of classes, $DF_(i)$ is the # of tweets in the training set containing the word i and $DF_i(c)$ is the # of tweets of the training set annotated with class c containing word i. This factor is used to weight the contribution $\omega_{i,d}$ of each term i in document d as (2):

$$\omega_{i,d} = TF_{i,d} \times log(\frac{N}{DF_{\mathbb{C}}(i)}) \times G_i \tag{2}$$

Where N is the number of tweets in the training set and the contribution $\omega_{i,c}$ of each term i in class c by replacing the word # of occurrences $TF_{i,d}$ by $DF_{i,c}$:

Baselines. We propose two baselines approaches. The first one consists in computing similarities between the tweet BoW and each class BoW as follows (3):

$$cos(d, c) = \frac{\sum\limits_{i \in d \cap c} \omega_{i,d} \times \omega_{i,c}}{\sqrt{\sum\limits_{i \in d} \omega_{i,d}^2 \times \sum\limits_{i \in c} \omega_{i,c}^2}} \tag{3}$$

The second one consists in training linear multi-class Support Vectors Machine [26] with the objective of classifying multiple classes. Classifiers have been trained with default parameters and the BoW vectorial representation of each tweet d (each term weight is computed as (2) but with DF_i instead of $TF_{i,d}$).

Multi-word Expression. In order to take into account the sequence of words in the tagging issue, we consider Conditional Random Fields(CRFs) [27], more exactly Linear CRFs. They represent log-linear models, normalized at the entire tweet level, where each word has an output class associated to it. Thus, CRFs can localize specific positions in tweets that carry information and highlight continuous contextual information. In this setup the probability between words and classes for the whole tweet (of N words) is defined as follows:

$$P(c_1^N|w_1^N) = \frac{1}{Z}\prod_{n=1}^{N}\sum_{m=1}^{M}\lambda_m \cdot h_m(c_{n-1}, c_n, s)) \tag{4}$$

Log-linear models are based on M feature functions h_m computed at each position from the previous class c_{n-1}, current class c_n and the whole observation sequence s (tweet). λ_m are the weights estimated during the training process and Z is a normalization term given by:

$$Z = \sum_{c_1^N}\prod_{n=1}^{N}\sum_{m=1}^{M}\lambda_m \cdot h_m(c_{n-1}, c_n, s)) \tag{5}$$

The tweets from the training set were used to train our CRF tagger with unigram (neighborhood window of length 2 around the current word) and bi-gram features. Then a CRF tagged each word in every tweets and decision for the final tweet's label is made by majority.

Lexical Context. RepLab test set vocabulary size is twice as big as the one of the annotated set. In order to reduce the impact of the information loss carried by out of vocabulary words (OOV), we project OOV into the known vocabulary in a Continuous distributed words representation [28] (considered as a generalization engine). We used a Word2Vec [29] model which is learn by a Skip-gram neural-network. This network try to maximize the following log probability [29]:

$$\frac{1}{N}\sum_{n=1}^{N}\sum_{-c<j<c, j\neq 0} log(\frac{exp(i_{w_{t+j}}^T o_{w_t}))}{\sum_{w=1}^{N} exp(i_w^T o_{w_t})}) \tag{6}$$

where N is the number of words in the training corpus, $w_0..w_N$ the sequence of training words, c the size of the context. Word2vec models where proved being able to capture syntactic and semantic relationship between words [29]. It allows us to measure similarity with simple geometric operations like sum and angle metrics. We trained a 600 dimensions,10 context windows, multilingual (English+Spanish) Skip-gram model over RepLab's background messages [3] which we added a large amount of easily available corpora[4]. This trained model

[4] enwik9, One Billion Word Language Modelling Benchmark, the Brown corpus, English GigaWord from 1 to 5, eswik, parallel es-en europarl

is then used as a generalization engine by other classifier i.e.: it finds for each OOV in the test sample the closest word in the Continuous distributed words representation which exists in the training vocabulary and has a sufficient purity as defined with (1).

4 Author Profiling

As far as the author of the messages plays a key role in determining the *Reputation Alerts* we have to profile authors. Besides their type, the number of comments and followers are also important aspects that determine the influence of an author in Twitter in a potential reputation-dangerous perspective. These tasks are usually addressed as Community Detection or Complex Network issues. This means that systems should define profile according to social meta-data (followers' graphs, numbers of favorites, followers or comments and so on). It then puts the need on complete relations graphs which may not be possible to extract in Twitter with private/deleted account and with queries limitations from Twitter's API. We understand *Author Profiling* as the following issue: using tweet contents that Twitter-users produced, systems have to reproduce experts' evaluations: ranking users according to their influence level and detecting the socioeconomic category users belong to.

4.1 Approaches

We investigate AP using different NLP-based profile representations. We made the same assumption of specific vocabulary that can differentiate opinion-makers from non opinion-makers[5] and users from separate socioeconomic categories. In our official submissions [23] we considered *k-NN* classification for *Influence Detection* and *Socioeconomic Categorization* (for this last task we also considered the Cosine described above (3)). We investigate the following user-profile definitions: *'User-as-document'* [11] and *'Bag-of-tweets'* (respectively noted *'UaD'* and *'BoT'*):

- *'UaD'* consists in merging all tweets from a profile to create one document and computing a similarity between each document and each class;
- *'BoT'* considers a binary classification problem for each tweet. Classification is achieved by counting the number of tweets tagged for the considered user.

The *'UaD' k-NN* consists in matching each user BoW to the most similar ones in the training set which are voting for the class they are belonging to according to the similarity index (here Jaccard). For the *'BoT'* Cosine, a user is deemed the belong to a socioeconomic category if a majority of his tweets are themselves considered to belong to this socioeconomic category. In both cases, ranking is achieved with the probability of being an *'Influencer'*.

[5] According to [9] influential Twitter authors in the economic domains considered in RepLab tend to be male in the 35-49 age range.

5 Experimental evaluation and results

5.1 Evaluation

We compare our proposal to RepLab baselines[6] and best submitted systems in both tasks. We report our results using RepLab official metrics. Although Accuracy (Acc) is a standard metric easy to understand, it has nevertheless a drawback when it comes to compare non-informative systems on unbalanced data-sets. RepLab organizers previously proposed a F-Measure (FM) based on Reliability and Sensitivity [30]. This metric compares the gold-standard with system produced priority relationship (in the case of the Alert Detection task). In addition, we compute an average F-Score (AvgF), based on Precision and Recall for each class which gives an overview of the system's ability to recover information from each class. Author Ranking is viewed as a Search problem, having the domain as query, systems have to return a ranking of the most relevant users. Evaluation is done according to Mean Average Precision (MAP) which compares ordered vectors based on a binary reference. Nevertheless with only two domains it is not possible to conclude that MAP improvements are significant.

5.2 Reputation Monitoring

We chose to tackle the classification issues with a global approach. Our experiments in using entity's or domain's specific training process shown no significant improvement.

Reputation Dimensions. Experts proposed the following Reputation Dimensions taxonomy: *Citizenship, Governance, Innovation, Leadership, Performance, Product&Services, WorkPlace* with an additional Undefined concept (see [4] for more details). As the *'Undefined'* class is excluded from the RepLab evaluation process, we first chose to investigate it as a filtering issue. That is to say, when systems are not able to significantly predict a dimension for a given tweet they tag this tweet as related to an *'Undefined'* Dimension. This experiment has shown no significant improvements for contextualized CRF (noted w/ Context in table 1), even if it simplifies the models' complexity (from 8 to 7 classes). Then, better than withdrawing the *'Undefined'* class in our evaluation (-U), we made additional experiments pulling back *'Undefined'* tweets (+U). Cosine is then significantly improved by the lexical context. All our proposal then performed competitively with respect to the best official submissions (noted Best_Acc and

[6] The organizers provided two baselines in the Reputation Dimension detection task and Author Categorization. A Naive one that assigns the most frequent class to each tweet. A ML-based classification using a linear SVM for each entity with Bag-of-Word's (BoW) binary representation. For the Priority Detection task the baseline consists in tagging the tweets of the test set with the label of the closest tweet (Jaccard similarity) in the training set. The Ranking baseline ranks authors by descending number of followers.

Best_F in table 1) and baselines. And that neither using generalization and considering the *'Undefined'* as part of the classification issue or not. Finally, the *'Undefined'* class shows interesting results. SVM as a discriminative method is the most perturbed by the novelty provided from this class. While additional context tend to reinforce Cosine and CRF robustness in generalizing new vocabulary, it has no effect on SVM performances.

Table 1: Dimensions detection performances ordered according to Accuracy(-U). Best performances are highlighted in bold. Statistical significant improvements (averaged across entities) over the SVM(-U) (two-sided pairwise t-test p < 0:05) are denoted *

Method	AvgF (-U)	Acc (-U)	AvgF(+U)	Acc(+U)
CRF w/ Context	.492	**.771***	.481	.761
CRF	.491	.769*	.483	.762
Cosine w/ Context	**.505**	.739	.494	.707
Cosine	.491	.736	.500	.693
SVM	.469	.732	.461	.679
SVM w/ Context	.468	.732	.456	.679
Best_Acc	.473	.731	-	-
Best_F	.489	.695	-	-
SVM Baseline	.38	.622	-	-
Naive Baseline	.152	.560	-	-

Priority Detection. We proposed during RepLab'2013 a kNN-based classification method [22] (noted Lia_Prio_5 in table 2) and obtained the best *FM (R,S)* reported up to our knowledge in this task. Other performances ranked with regards to FM are noted in Table 2. Both SVM and Cosine approaches are competitive according to *Acc* but their *AvgF* remain lower and the Cosine even stays lower than the Replab2013 baseline according to FM (R,S). Given the relatively limited number of *'Alerts'*, an alternative evaluation reconsidering the classification issue as search problem (ranking) should provide interesting additional information.

Table 2: Priority detection performances ordered by F-Measure (R,S).

Method	AvgF	Acc	FM (R,S)
Lia_Prio_5	**.571**	.636	**.335**
SVM	.563	.644	.304
CRF	.554	.633	0.318
SVM w/ Context	.564	**.645**	.304
CRF w/ Context	.551	.631	0.318
Baseline	.512	.570	.274
Cosine w/ Context	.562	.634	.260
Cosine	.561	.633	.260

Our experimental evaluations establish that tweet lexical content is sufficient for simple ML approaches to tackle the tasks of identifying the reputation alerts and dimensions. Our experiments with generalization shown that lexical context can be useful and efficient in dimension assignment but not for reputation alerts.

5.3 Author Profiling

We chose to process English and Spanish messages separately to reduce the models complexity for the *'UaD'* approach. We supposed that profiles have particular influence or socioeconomic characteristics in their vocabulary regardless the domain they are mainly associated with.

Author Ranking. As the experts did not rank the authors, we first have considered a binary classification problem for each author. The ranking can be tackled as a post processing applied on the binary classification output [23]. We have chosen to rank the authors according to their probability of being *'Influencers'*. As it can be seen in table 3, our proposal (noted Lia_AR_1) got an Average MAP under the best system, but it was globally better than the Baseline approach. In the Banking domain, which seems to be a difficult one, Lia_AR_1 performed better than both of them.

Since this system was optimized on a development set in order to maximize the AvgF and not the MAP (used for evaluation), there was clearly some room for improvement. We then chose to estimate the parameter values of a *k-NN* for each language with the purpose to maximize the *MAP*[7] on the development set. For instance, with k=16 for English, (17 for Spanish) on the development. It can been observed in table 3 (row LIA_NEW) that on the test set, the results are better than the ones obtained by all submitted systems. It is nevertheless impossible to verify that improvements are statistically significant.

Table 3: Author Ranking performances ordered according to Average MAP.

Method	Automotive	Banking	Average MAP
Lia_NEW (UaD)	**.764**	**.652**	**.708**
Best	.721	.410	.565
Lia_AR_1 (UaD)	.502	.450	.476
Baseline	.370	.385	.378
Cosine (BoT)	.207	.194	.200

Author Categorization. Organizers reported that only one approach [23] (noted Lia_AC_1 in the left part of table 3) performed as well as *'most frequent class'* and ML SVM baselines according to *Acc*. Cosine_RA uses a re-affectation post-process over the Cosine output (described in the next section) in order to fit class distribution of the training set but it shown no performances improvements. The SVM baseline reaches the best *AvgF* and stay far above all proposals when considering the *'Undecidable'* class as part of the evaluation process. When we ignore the *'Undecidable'* (the right part of table 3) from the evaluation process the re-affectation post-process allows small *AvgF* improvements with limited losses in *Acc* but it remains unsatisfactory. Author Categorization is still an open problem. The availability of more data will surely allow to propose a deeper results analysis.

[7] This way, there is no need to introduce any offset or to penalize a class as done previously in [23].

Table 4: Author Categorization performances ordered by Average Accuracy.

Method	Average Acc	AvgF	Method	Average Acc	AvgF
Lia_AC_1 (UaD)	.471	.269	Cosine (BoT)	**.486**	.244
Baseline-SVM	.460	**.302**	Cosine_RA (BoT)	.481	.294
MF-Baseline	.435	-	Lia_AC_1 (UaD)	.393	.253
Cosine (BoT)	.346	.185			
Cosine_RA (BoT)	.341	.221			

5.4 Classes distribution issue and perspectives

With regard to a large variety of label distribution in the *Author Categorization* and *Dimensions Detection* training sets, we decided to have a harmonization post-process of our systems output. For each output the post-process consists in considering the second hypothesis of the system (fill small classes despite having a better confidence in a bigger class) in the following case:

- The best hypothesis is an over-populated class [8]
- The second hypothesis is an under-populated class
- The score differential between the two hypotheses is not significant.

Another approach, taking root in the field of game theory, could also be considered. In [31], the author applies a matching game algorithm to a ranking problem in the context of movie recommendations. This method lie in the fact that both movies and users have preferences and both of them are involved in the recommendation process. In other words, the system does not simply recommend the best movie based on the user's preferences, but also takes into account the movie point of view, by somehow selecting the best candidates. If we transpose this idea to our classification problem, we could consider a matching game in which the players would be authors (or tweets) and classes. Each author would have its preferred class and each class its preferred author. More precisely, both would have a list ordered by preferences which means regarding each class we would be able to select the most representative tweets.

6 Conclusions

RepLab contests allowed us to perform a large number of experiments with state-of-the-art evaluation. Our experimental evaluations establish that discriminating textual features inferred from expert assessments coupled with simple ML approaches is sufficient to expand these feedback to unlabeled data. However, while the results remain lower in term of *Author Profiling*, they prepare the ground for further experiments using additional data.

In future work, we plan to examine relations between classes and in a broader sense tasks to discover latent hierarchies. Since *Lexical Context* provided interesting results we also intend to study an interesting *Lexical Expansion* simulating, *Active Learning* over non-annotated provided tweets. In this way, tweets

[8] The notion of over or under population is considered with regards to the class distribution in the training set.

which do not correspond to expert stresses could be filtered or manually checked before being re-injected as supplementary training material. Since the mass of data has caused many problems, we will consider automatic summarizations of tweet clusters and user profiles to reduce class spaces and perform a more rapid classification.

6.1 Acknowledgment

This work is funded by ImagiWeb ANR-2012-CORD-002-01. Thanks to Judith Cuinier for proofreading this paper.

References

1. O'Connor, B., Balasubramanyan, R., Routledge, B.R., Smith, N.A.: From tweets to polls: Linking text sentiment to public opinion time series. ICWSM **11** (2010)
2. Peetz, M.H., Spina, D., Gonzalo, J., De Rijke, M.: Towards an active learning system for company name disambiguation in microblog streams. In: CLEF 2013
3. Amigó, E., De Albornoz, J.C., Chugur, I., Corujo, A., Gonzalo, J., Martín, T., Meij, E., De Rijke, M., Spina, D.: Overview of replab 2013: Evaluating online reputation monitoring systems. In: CLEF 2013
4. Amigó, E., Carrillo-de Albornoz, J., Chugur, I., Corujo, A., Gonzalo, J., Meij, E., de Rijke, M., Spina, D.: Overview of replab 2014: author profiling and reputation dimensions for online reputation management. In: CLEF 2014
5. Villena Román, J., Lana Serrano, S., Martínez Cámara, E., González Cristóbal, J.C.: Tass-workshop on sentiment analysis at sepln. (2013)
6. Hangya, V., Farkas, R.: Filtering and polarity detection for reputation management on tweets. In: CLEF 2013
7. Gârbacea, C., Tsagkias, M., de Rijke, M.: Detecting the reputation polarity of microblog posts, ECAI (2014)
8. Spina, D., Gonzalo, J., Amigó, E.: Learning similarity functions for topic detection in online reputation monitoring. In: Proc. of the 37th SIGIR conference on Research & development in IR. (2014)
9. Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daelemans, W.: Overview of the author profiling task at pan 2014
10. Sandoval-Almazan, R.: Using twitter in political campaigns: The case of the pri candidate in mexico. International Journal of E-Politics (IJEP) **6** (2015)
11. Kim, Y.M., Velcin, J., Bonnevay, S., Rizoiu, M.A.: Temporal multinomial mixture for instance-oriented evolutionary clustering. In: Advances in IR. (2015)
12. Velcin, J., Kim, Y., Brun, C., Dormagen, J., SanJuan, E., Khouas, L., Peradotto, A., Bonnevay, S., Roux, C., Boyadjian, J., et al.: Investigating the image of entities in social media: Dataset design and first results. In: LREC 2014
13. Pla, F., Hurtado, L.F.: Political tendency identification in twitter using sentiment analysis techniques. In: Proc. of COLING. (2014)
14. Vilares, D., Hermo, M., Alonso, M.A., Gómez-Rodrıguez, C., Vilares, J.: Lys at clef replab 2014: Creating the state of the art in author influence ranking and reputation classification on twitter. In: CLEF 2014
15. Gobeill, J., Gaudinat, A., Ruch, P.: Instance-based learning for tweet categorization in clef replab 2014. In: CLEF 2014

16. Berrocal, J.L.A., Figuerola, C.G., Rodríguez, Á.Z.: Reina at replab2013 topic detection task: Community detection. In: CLEF 2013
17. Ramírez-de-la Rosa, G., Villatoro-Tello, E., Jiménez-Salazar, H., Sánchez-Sánchez, C.: Towards automatic detection of user influence in twitter by means of stylistic and behavioral features. In: Human-Inspired Computing and Its Applications
18. McDonald, G., Deveaud, R., McCreadie, R., Macdonald, C., Ounis, I.: Tweet enrichment for effective dimensions classification in online reputation management. In: Ninth International AAAI Conference on Web and Social Media. (2015)
19. Qureshi, M.A., ORiordan, C., Pasi, G.: Exploiting wikipedia for entity name disambiguation in tweets. In: NLP and Information Systems. (2014)
20. Cossu, J.V., Bigot, B., Bonnefoy, L., Senay, G.: Towards the improvement of topic priority assignment using various topic detection methods for e-reputation monitoring on twitter. In: NLP and Information Systems. (2014)
21. Cossu, J.V., San-Juan, E., Torres-Moreno, J.M., El-Bèze, M.: Automatic classification and pls-pm modeling for profiling reputation of corporate entities on twitter. In: Natural Language Processing and Information Systems. (2015)
22. Cossu, J., Bigot, B., Bonnefoy, L., Morchid, M., Bost, X., Senay, G., Dufour, R., Bouvier, V., Torres-Moreno, J., El-Bèze, M.: Lia@replab 2013. In: CLEF 2013
23. Cossu, J.V., Janod, K., Ferreira, E., Gaillard, J., El-Bèze, M.: Lia@ replab 2014: 10 methods for 3 tasks. In: CLEF 2014
24. Sparck Jones, K.: A statistical interpretation of term specificity and its application in retrieval. Journal of documentation **28** (1972) 11–21
25. Torres-Moreno, J., El-Beze, M., Bellot, P.: Bechet, opinion detection as a topic classification problem in in textual information access. chapter 9 (2013)
26. Crammer, K., Singer, Y.: On the algorithmic implementation of multiclass kernel-based vector machines. Journal of Machine Learning Research **2** (2002) 265–292
27. Lafferty, J., McCallum, A., Pereira, F.C.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. (2001)
28. Bengio, Y., Ducharme, R., Vincent, P.: A neural probabilistic language model. Journal of Machine Learning Research **3** (2003) 1137–1155
29. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. (2013) 3111–3119
30. Amigó, E., Gonzalo, J., Verdejo, F.: A general evaluation measure for document organization tasks. In: Proc. of the 36th SIGIR conference on Research & development in IR. (2013)
31. Gaillard, J.: Recommendation Systems: Dynamic Adaptation and Argumentation. PhD thesis, University of Avignon, France (2014)