Sentiment Estimation on Twitter

Giambattista Amati, Marco Bianchi, and Giuseppe Marcone

Fondazione Ugo Bordoni, Rome, Italy
gba@fub.it, mbianchi@fub.it, gmarcone@fub.it

Abstract We study the classifier quantification problem in the context of the topical opinion retrieval, that consists in estimating proportions of the sentiment categories in the result set of a topic. We propose a methodology to circumvent individual classification allowing a real-time sentiment analysis for huge volumes of data. After discussing existing approaches to quantification, the novel proposed methodology is applied to Microblogging Retrieval and provides statistically significant estimates of sentiment category proportions. Our solution modifies Hopkins and King's approach in order to remove manual intervention, and making sentiment analysis feasible in real time. Evaluation is conduced with a test collection made up of about 3,2M tweets.

1 Introduction

Sentiment analysis for social networks is one of the most popular and mature research problem in Information Retrieval and Machine Learning. Several companies already provide services aimed to detect, estimate, and summarize opinions of social network users on different topics. As a consequence, a "gold rush" is started on finding scalable solutions to perform real-time analysis on huge volumes of data.

In general, real-time content analysis should accomplish different tasks: distilling high-quality samples about populations under investigation, labeling samples by concepts and taxonomies, and providing accurate estimation of sizes and proportions of category populations. Although accurate classification of single documents and high precision retrieval are important desiderata of real time content analytics, decision making often requires quantitative analysis, that consists in computing accurately the estimates of populations size or the proportion of individuals that fall into a predefined category. For example, what is important in sentiment analysis or topical opinion retrieval [1] is how the entire population distribute by sentiment polarities relatively to different topics, possibly at certain instant of time or showing temporal trends for these distributions.

We propose a methodology that provides statistically significant estimates of sentiment category proportions in social science, and in particular in the Microblogging Retrieval that is for the result set of tweets of a topic. In particular, we revisit the *classifier quantification* problem in the context of topical opinion retrieval. Classifier quantification was introduced earlier in the 70s in epidemiology [2], an area in which - similarly to social science - the quantities of interest are at aggregated level. Classifier quantification was later reconsidered in Machine Learning [3].

It is worth noting that, in principle, the classifier quantification can be done applying two components in sequence: after retrieval, the result set of the query is passed to a sentiment classifier; then, after classification, one can count individuals in each category set. However, according to Pang et al. [4] who compared more than twenty methods for sentiment analysis, classification performance does not go beyond 83% of accuracy, that it is not enough to compute a statistically significant estimate of categories proportions.

To remedy classifier inaccuracy and predict the actual opinion category size, Forman introduces a "classify and count" (CC) strategy [3,5], that consists in counting the individuals of each category set after classification ($|\hat{D}_k|$), and then adjusting the observed proportions with the classifier error rates $P(\hat{D}_j|D_k)$ with $j \neq k$, that are obtained after training the classifier:

$$P(\hat{D}_j) = \sum_k P(\hat{D}_j|D_k) \cdot P(D_k)$$

The actual categories proportions $P(D_k)$ are then solutions of all k (adjusted-CC) linear equations. Since a sentiment classifier is trained independently from queries, and the collection C contains millions of documents, $P(\hat{D}_j|D_k)$ can be observed only on a small sample C' of the collection. To accept these adjusted-CC solutions as correct sentiment category proportions among a subset R_q of relevant documents for a query q, we need to assume that $P(\hat{D}_j|D_k)$ are conservative on the subset of relevant documents of the query, even if the intersection $R_q \cap C'$ might result empty. Alternatively, we need to introduce relevance as an extra category (actually irrelevance as a mutually exclusive extra category), and train the classifier on each query. The conservative assumption is unrealistic, because there always is a bias of sentiment distribution between relevant and non-relevant information, and sentiment analysis quantifies such a bias. On the other case, training the classifier on each query would be time consuming and requires a manual intervention.

Notice that since the adjusted-CC approach uses misclassification proportions $P(\hat{D}_j|D_k)$, it works with any classifier accuracy. What is important for a statistically significant prediction is how large the random sample of the query is. Therefore, if the training set is a large random sample of the query population, then a query-by-query approach becomes *de facto* a manual evaluation of sentiment. Although the retrieving process is fast, classification task is instead time consuming so that the adjusted-CC approach is not feasible for real-time analytics.

In topical opinion retrieval, the strategy of filtering by sentiment the relevance ranking always degrades retrieval performance with respect to other reranking strategy [6,7], due to the negative effects of the removal of the misclassified relevant information [8].

We instead follow more closely the model proposed by Hopkins & King [9] and adapt it to retrieval. Hopkins & King get rid of the classifier and propose a "word profiles counting" approach among a random sample of evaluated documents. Among $S = 2^{\mathbf{V}}$ possible subsets (profiles) of the set of words they choose a random sample $S' \subset S$ and count the occurrences of profiles $s \in S'$ occurring in the true classification D_k:

$$\mathrm{P}(s \in S') = \sum_j \mathrm{P}(s \in S'|D_j) \cdot \mathrm{P}(D_j) \tag{1}$$

Then $\mathrm{P}(D_j)$ are obtained as regression coefficient of a linear regression set of equations.

The advantage of Hopkins & King's methodology is that of estimating $\mathrm{P}(D_j)$ without the intermediate step of computing the individual classifications. Unfortunately for each query, a manual classification is required for a quite large number of individuals of the population. Hopkins & King's methodology is thus manual and provides only an automatic smoothing technique to make the manually labeled proportions statistically significant.

We now adapt Hopkins and King's basic model to a topical opinion retrieval. The main difference with Hopkins and King's methodology is the use of a set S_k of learned (biased) features for the set S' (and not a random word profile sample), one for each category D_k. Also we do not count the features in the true category distribution of a query, but we use an information theoretic approach that instead counts the number of bits necessary to code all the features S_k that occurs in the result set of a query. The code is obtained with respect to the true distribution of a query-independent training set. We then assume that all these aggregated numbers of bits are linearly correlated to the number $\mathrm{P}(\hat{D}_k|D_j, q)$. In such a way we avoid the classification step as in Hopkins and King's but we also avoid to use a manual evaluation for the specific query q.

In the next sections we present this approach in order to handle arbitrary queries without training on single queries.

2 Revisiting Hopkins and King's Model

First we exploit the theorem of total probability:

$$\mathrm{P}(\hat{D}_k|q) = \sum_j L(\hat{D}_k|D_j, q) \cdot \pi(D_j|q) \tag{2}$$

where $\hat{D}_k$ is the k-th category set with the chosen classifier, D_j is the true j-th category set, q the query, $L(D_j|D_k, q)$ is the *likelihood*, and $\pi(D_k|q)$ is the unknown *prior* on the set of categories (the actual category proportions). The non diagonal elements $L(\hat{D}_i|D_j, q)$ of the matrix L, with $i \neq j$, are the errors due to all misclassified individuals.

> The estimation problem can be thus restated to how to find estimates $\hat{\pi}(D_k|q)$ for $\pi(D_k|q)$ that predict the proportions in the population for the categories D_k.

We now define population size prediction in terms of features of a textual classifier. A textual classifier $\hat{D}$ can be defined as a set of features S (or a weighting function from $\hat{D}$ to S) that assigns individuals to single categories. For text classifiers $\hat{D}$, S in particular is a set of words, each word having a probability $L(s \in S|D_i, q)$ of occurrence in the i-th category. S can be thus chosen as a set of words *spanning* over the population Ω of individuals, and thus S is simply a function of the classifier $\hat{D}$. For weighting function $w^q_{s,i}$, such as the SVM classifier or IR weighting models such as standard tf-idf model or other topic-specific lexicon [10,11], we can make the assumption:

$$L(s \in S|D_i, q) = \alpha \cdot w^q_{s,i} \tag{3}$$

where the parameter α is just a normalizing factor. The theorem of total probability spanning over a set S of features is

$$\mathrm{P}(s \in S|q) = \sum_{i=1}^{n} L(s \in S|D_i, q)\pi(D_i|q) \tag{4}$$

where $L(s|D_i, q)$ is the *likelihood* and $\pi(D_i|q)$ is the *prior* over the set of categories.
We may rewrite the Equality (4) in matrix form:

$$\underset{|S|\times 1}{\mathrm{P}(S|q)} = \underset{|S|\times|\mathcal{D}|}{L(S|D, q)} \cdot \underset{|\mathcal{D}|\times 1}{\pi(D|q)}$$

Our problem formulation then can be restated as the problem of finding best fitting values $\hat{\pi}(D_i|q)$ for the vector $\pi(D_i|q)$ in a set of linear equations.
To learn the likelihood matrix $L(\hat{D}|D, q)$ on the entire population one can use instead a likelihood matrix L_{X_q} over a sample X_q of retrieved tweets with respect to the query q and error ϵ $(L \sim L_{X_q})$, that is

$$\mathrm{P}(\hat{D}|q) = L_{X_q}(\hat{D}|D, q) \cdot \pi(D|q) \tag{5}$$

Passing through the set S of spanning features

$$\underset{|S|\times 1}{\mathrm{P}(S|q)} = \underset{|S|\times|\mathcal{D}|}{L_{X_q}} (S|D, q) \cdot \underset{|\mathcal{D}|\times 1}{\pi(D|q)}$$

we get the latent regression matrix $\underset{|\mathcal{D}|\times|S|}{L'_{X_q}}$ such that

$$\underset{|\mathcal{D}|\times 1}{\pi(D|q)} = \left(\underset{|\mathcal{D}|\times|S|}{L'_{X_q}} \cdot \underset{|S|\times|\mathcal{D}|}{L_{X_q}(S|D, q)} \right)^{-1} \cdot \underset{|\mathcal{D}|\times|S|}{L'_{X_q}} \cdot \underset{|S|\times 1}{\mathrm{P}(S|q)}$$

We may use the linear regression with the two matrices L_{X_q} and P. Therefore the estimates $\hat{\pi}(D_k|q)$ of $\pi(D_k|q)$ becomes the coefficient for the k-th category of the linear regression and provide the estimated proportion for the category D_k.
The main computational cost of this methodology is that S should be learned by the chosen classifier $\hat{D}$, that is S is a function of $\hat{D}$ and the topic q.
A sentiment textual classifier has at least the following categories:

$$\{S^+, S^-, S^{NO}, S^{Mix}, S^{NR}\}$$

In presence of a query or a set of queries q, S^{NR} contains all non relevant elements, S^{NO} relevant but without opinions, S^+ (S^-) relevant and strictly positive (negative) opinions, whilst the individuals containing mixed opinions fall into the remaining category S^{Mix}. To learn S, and thus to build L_{X_q}, a manual inspection is required and this is the main drawback of this methodology to perform any real time analytics. Hopkins & King suggest to use a random sample of word profiles and manually annotate about 500 blog posts to achieve 99% of statistical significance of the estimates $\hat{\pi}(D_i|q)$.
In case of an adjusted-CC approach with individual classifiers, such as SVM, there is a highly time intensive tuning phase. Hopkins and King reports that each run of their estimator based on SVM took 60 seconds of computer time, and a total of five hours for 300 bootstrapped runs, for a collection of 4k blog posts for a total number of 47k tokens.

3 Real time analytics

We now want to get statistics on the sentiment polarity for the entire population relatively to an arbitrary query q at retrieval time.
$L_{X_q}(s_i \in S|D_k, q)$ can be interpreted as the percent of times the sentiment term s_i occurs in the set X_q of evaluated tweets in D_k and retrieved by the query q, and $\mathrm{P}(s_i \in S|q)$ is the percent of times the sentiment term s_i occurs in the retrieved set. Once we have the evaluation on the query q the input matrices of L_{X_q} and P can be computed quite fast. A sentiment dictionary even when containing many thousand of elements is submitted to the system as a query and would be feasible to build the input matrices in seconds, or much less within a distributed system. However regression algorithm would require additional computation time, even though it would require still reasonable computational time in presence of a relatively small result set.

3.1 Cumulative scoring assumption

We now describe in details the approach. In the training phase we pool all relevant documents irrespective to the query by which they were retrieved and then we select the opinionated documents from the rest. Therefore we formed two collections the sentiment sample included into a document sample. The reason we use only relevant information is to use a fully evaluated collection, and thus reject from the sentiment dictionary words that occur randomly in both collections. The collection of relevant documents provides a prior distribution for the terms s (see π_s below). We then use the features $s \in S^i$ of a classifier learned by the sentiment category D_i. We apply the linear regression algorithm of Equation (5) and learn the regression coefficients. Each coefficient will be then multiplied by the sum of sentiment scores of the retrieved documents to obtain the number of tweets in a category for any new query q.

Training phase In the training phase we make the following assumptions:

- (*Multiple binary dictionaries reduction*) We first assume a binary classifier for each category k. In particular for the sentiment analysis, we learn from only the positive and negative result sets, that is producing two distinct dictionaries S^+ and S^-. We have thus two Equations (4), each therefore restricted to one of the two chosen categories. The likelihood for that category k is thus: $L_X(s_i \in S^k|D_k, q)$.
- We use information theoretic arguments to compute $L_X(S^k|D_k, q_j)$ (but it is not necessarily a theoretical limit since any other score weighting formula or classifier can be similarly used here). We pool the result sets D^k_Q for a category k from a set of training queries Q and extract frequencies of terms for that category S^k, and build the classifier for the k-th category

  $$I(s \in S^k) = -\log \mathrm{P}(s|\pi_s, D^k_Q)$$

  which is given by a binomial P with prior distribution π_s and frequencies in D^k_Q.

- The proportion of documents $L_{X_q}(\hat{D}^k|D^k, q)$ falling in a given category with respect to the result set of a query q is proportional to the amount of sentiment information for that category in the result set of that query [12], that is

$$I(S^k|q) = \sum_{s \in S^k, d \in X_q} - \log \mathrm{P}(s|\pi_s, D^k_Q)$$

- We train and test the classifier with linear regression on a set of queries Q and establish how many queries are necessary to learn the estimate $\hat{\pi}(D_k)$;
- misclassifications $L_X(S \neq S^k|D_k, q_j)$ and $L_X(S^k|D \neq D_k, q_j)$ are not computed but recovered by learning distinct regression coefficients $\hat{\alpha} \cdot \hat{\pi}(D^k)$:

$$\mathrm{P}(\hat{D}^k|q) = \alpha \cdot I(S^k|q) \cdot \pi(D^k) \text{ for a set of queries } q \in Q \tag{6}$$

where $\mathrm{P}(\hat{D}^k|q)$ is the percent of observed individuals in the category D^k with respect to the query q. The errors in prediction between observed and predicted are given by the residuals $\mathrm{P}(\hat{D}^k|q) - \hat{\alpha} \cdot I(S^k|q) \cdot \hat{\pi}(D^k)$.

Retrieval phase In the retrieval phase we assume that the number of tweets in the category k with respect to any query q is:

$$|D_k| = \hat{\alpha} \cdot I(S^k|q) \cdot \hat{\pi}(D^k) \tag{7}$$

and proportions are given by $\frac{|D_k|}{|C_q|}$ where $|C_q|$ is the result set size of the query. Notwithstanding that the ranking will contain both irrelevant and misclassified tweets for the k-th category that have a positive score $I(s \in S^k|q)$, the predicted number of relevant tweets falling into the k-th category is very close to the actual number.

4 Experimentation

4.1 Evaluation measures

The evaluation of category size and category proportions for sentiment analysis is not trivial. Forman suggests to use the Kullback-Leibler distance between category distribution of CC-adjusted values and true values in the test set to measure effectiveness of the prediction, irrespective of the population size. This measure however though very simple does not comply with evaluation of quantification in social science or in statistics where residuals between predicted and observed values are measured. Also KL measures the distance between two distribution irrespective to the test set size. More precisely, to validate classifier effectiveness for individual classification an average value of different test sets smaller than their training sets is usually used, for example through a k-fold cross validation, so that the accuracy (which is a percentage of decision successes of the classifier) may not be statistically significant with respect to the actual population size. For example if a query has millions of relevant tweets and the evaluation is done on a few hundreds of them, then the true accuracy can falls into a very large confidence interval of the observed accuracy at 95% or higher confidence level. As an example, the third row of Table 2 shows a prediction of 45% positives within a population of 251K tweets, but the manual inspection on a sample shows 41% which falls into the confidence interval $[37.1\%, 52\%]$ at the 95% confidence level. In such a case, it is required more evaluation of tweets to reject or accept that proportion prediction.

In addition, when quantification is not performed with a variant of a CC approach, such as our method which is not based on individual classification, then we cannot compute the accuracy of the classifier but may only study the residuals between the predicted and the observed values.

Here, we use the R-squared (sum of the squares of the residuals) regression analysis to assess the goodness of fit between observed and estimates of the category sizes, the number of queries (minus 1) being the degrees of freedom.

Notice that, when individual classifier is used in quantification, the residual of the aggregated statistics is

$$Obs^{+} - Pred^{+} = tp + fp - (tp + fn) = fp - fn = -(Obs^{-} - Pred^{-})$$

Therefore, the misclassification errors in quantification or in aggregated classification are less severe than in individual classification, because the missing counting of the false negatives is partially balanced by the counting of the false positives.

This observation does not complete evaluation issues. Relevance and sentiment polarity cannot be split in the evaluation. Therefore the true distribution of a test set can be only given for a sample of the population of a specific user query. However, the sentiment polarity of a non-relevant retrieved document always contribute to size prediction, and for difficult queries there are many non-relevant documents. How the difficulty of a query impacts sentiment category size prediction must be studied, though we may conjecture that the sentiment noise brought by irrelevant data maximizes the entropy of the probabilities over the sentiment vocabulary in the result set. The entropy maximization smooths the predicted sizes of a polarity skewed set of a large result set towards the polarity means in the collection, that is predicted sizes converge to milder values.

As final remark, in classification it is also assumed that the test set was randomly built from the entire population. In general this is not the case, especially in order to remove imbalance between positive and negative category sizes. On the contrary, in IR we easily have imbalance due to the sparsity of the terms over the collection. We may be accurate on some queries less on others so that the set of predictions $x_q, q \in Q$, must be assessed by standard statistical tests, that is studying the distribution of residuals to show how good was the fit (R-squared) or to single out possible query outliers (Cook's distance or Normal Q-Q distributions etc.).

As a consequence of the above considerations, we have to distinguish proportion estimates from population size estimates, because they can largely vary. When the result set of a query is very small then proportions can be meaningless and not significant, whilst for large populations with small evaluated samples population size prediction can fall into a very large confidence level.

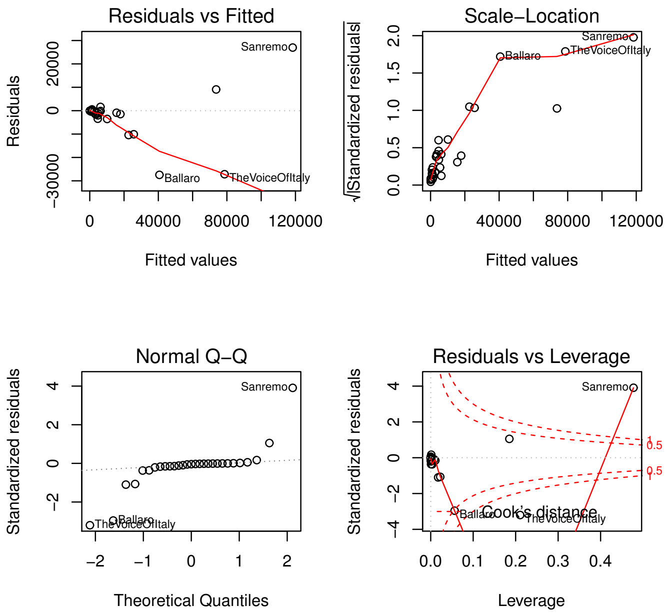

Figure 1. The linear regression coefficient estimate of positive category is statistically significant (p-value <2e-16). Multiple R-squared is 0.9199, and F-statistic has a p-value: < 2.2e-16. Positive outliers are “Sanremo”, “Ballarò” and “The Voice of Italy”.

4.2 Benchmark description

TREC Twitter Collection Tweets2011 is the only publicly available very large collection to test retrieval performance of models, but there is not yet a test collection on Twitter to conduct sentiment analysis. Obviously, nothing exists for the Italian language. However, Twitter's policy for distribution requires the redistribution of only tweet IDs (or user IDs) not their content, and this restriction together with the limitation of a few hundreds for the maximum number of API requests by hour (by the GET statuses/show/:id or users/lookup :id methods) makes the actual distribution of a very large collection prohibitive. In order to test our methodology, especially for the Italian language, we have thus conducted a sentiment analysis campaign about the Italian TV broadcasting. From December 2012 up to April 2013, we collected about 3,2 million tweets related to 30 TV programs. More than 6300 tweets have been evaluated by a team of 5 assessors. For each TV program a random selection of tweets have been manually annotated in terms of:
Relevance, that is:

- *highly relevant* (R^+), if the main topic of the tweet was about the TV program itself;

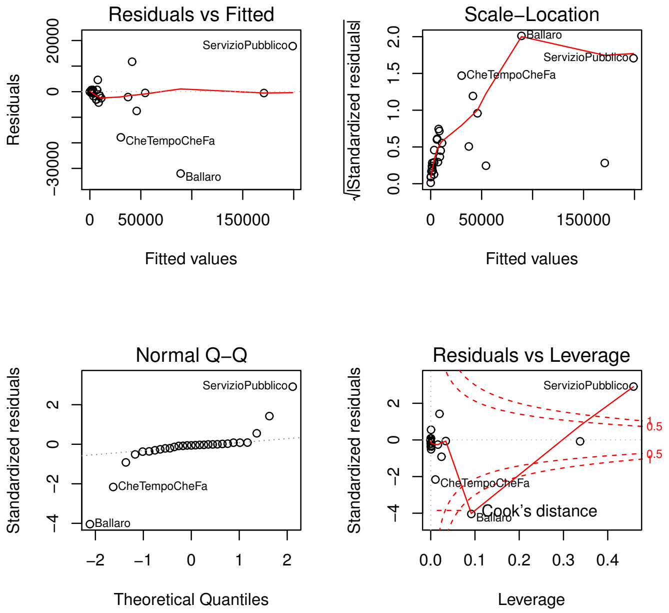

Figure 2. The linear regression coefficient estimate of negative category is statistically significant (p-value <2e-16). Multiple R-squared is 0.9781, and F-statistic: has a p-value: < 2.2e-16. Negative outliers are "Servizio Pubblico", "Ballarò" and "Che tempo che fa".

– *relevant* (R), if the main topic of the tweet was related to the TV program, such as a guest or a topic discussed during the TV program;
– *non-relevant* (NR), otherwise.

Opinion. Each relevant tweet then was evaluated as:
- *positive* (O^+), if containing a positive opinion;
- *negative* (O^-), if containing a negative opinion;
- *mixed* (Mix), if containing both positive and negative opinions;
- *neutral* (NO), if not containing opinions;
- *other* (NC), otherwise.

The results on the annotation activity are summarized in Table 1.

4.3 Results and discussion

Although, the test phase is still preliminary, because of the small number of training queries and number of evaluated tweets for each query to learn dictionaries and test the classifiers, the 5-fold cross validation worked well (see Figure 3). To notice that the amount of overall positive information in the training set is much less than the negative one, and therefore a number of 30

Relevance				Sentiment					
R^+	R	NR	$Total$	O^+	O^-	Mix	NO	NC	$Total$
787	5518	3910	10215	1358	2293	382	1959	313	6305

Table 1. Negativity prevails on TV tweet comments. We have imbalanced collection of data to learn sentiment dictionaries.

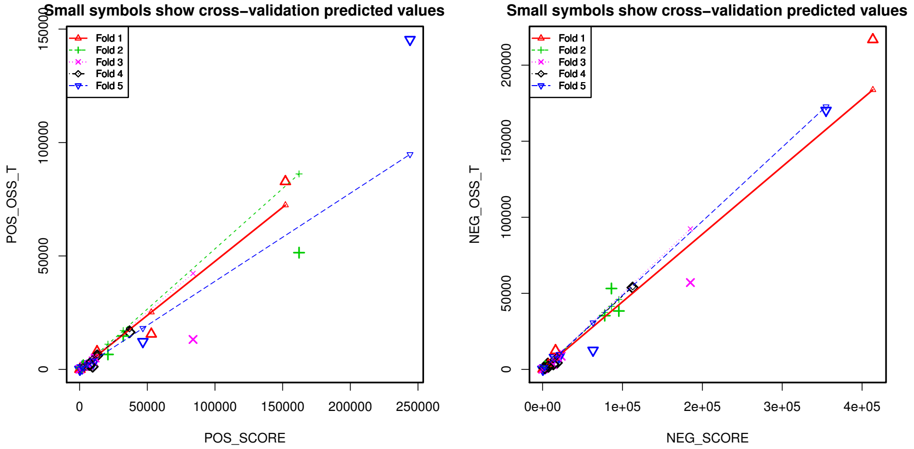

Figure 3. 5-Fold cross-validation for positive and negative data. The ANOVA table of the data has the statistics b$F = 3018$ for negative and $F = 322$ for positive with 28 degrees of freedom and p-value $Pr(> F) = 2 \cdot e^{-16}$ which provides a statistically significant predictive model.

queries for both testing and training is extremely small to learn a satisfactory vocabulary for positive polarity.

The quantile plots of Figures 1 and 2 (the Q-Q plots) show a normal distribution around a line with some outliers in the right upper corner. These outliers are caused by difficult queries, that is the queries with many relevant documents. For such queries the cumulative sentiment score becomes a less precise indicator for the prediction of the size of the category population, especially when there is a large bias of the positive (negative) opinions from their mean. Indeed, the outliers of the Figures 1 and 2 are those that receive the highest number of tweets with respect to the rest of the training queries. Also they have a relatively larger number of negative and positive tweets respectively with respect to others (the TV talk show called “Servizio Pubblico” has a very few percent of positive tweets, whilst the singer contest called “Sanremo” has a percent number of positive tweets that is larger than the mean 27.5%). In our opinion big national and international events (when the result set is very large) may require a specific classifier and not necessarily a linear regression classifier. Further investigation on how relevance (the number of irrelevant

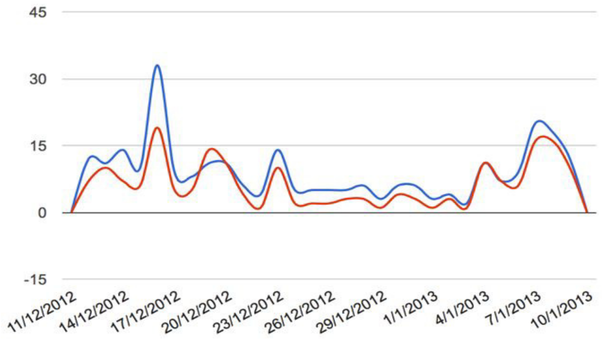

Figure 4. Comparison of the number of positive and negative tweets at run-time.

tweets in the results set) affects performance of sentiment predictions is thus required.

5 Conclusions

We have shown that sentiment estimation can be conducted in real time. For an arbitrary query, plot on Figure 4 shows the projected number of positive and negative tweets at different instant of time. At the moment, the two classifiers (positive and negative) are independent and do not interact. It can thus happen that sentiment quantification can produce inconsistent statistics when category proportions are compared.Relaxing the multiple binary dictionaries reduction assumption that handles separately positive and negative classifiers is thus required.

	Ret	POS	POS	NEG	NEG
		(pred)	(obs)	(pred)	(obs)
Sanremo	825,596	23.2%*	[29.3%, 38.4%] (33.7%)	33.4%	[34.9%,44.3%] (39.5%)
Servizio Pubb.	659,264	18.7%*	[21.0%, 31.3%] (25.9%)	50.5%*	[62.0%, 72.9%] (67.7%)
VoiceOfItaly	251,296	45.7%	[37.1%, 52.3%] (41.1%)	26.7%*	[28.5%, 43.1%] (30.7%)
Ballarò	344,683	19.9%*	[8.4%, 15.3%] (11.4%)	43.1%*	[44.1%, 54.8%] (49.4%)
CheTempoCheFa	134,455	27.3%	[28.8%, 38.6%] (33.5%)	34.3%	[29.5%, 39.4%] (36.6%)

Table 2. Big events require the evaluation by sentiment of a very large sample of relevant tweets to reduce the confidence interval (within square brackets). Outliers (denoted by a *) have indeed very large results sets.

Acknowledgment

Work carried out under Research Agreement with Almawave.

References

1. Ounis, I., De Rijke, M., Macdonald, C., Mishne, G., and Soboroff, I. Overview of the TREC-2006 Blog Track. In *In Proceedings of the Text REtrieval Conference (TREC 2006)* (2006), National Institute of Standards and Technology.
2. Levy, P. S., and Kass, E. H. A three population model for sequential screening for bacteriuria. *American Journal of Epidemiology 91* (1970), 148–154.
3. Forman, G. Counting positives accurately despite inaccurate classification. In *ECML* (2005), pp. 564–575.
4. Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up?: sentiment classification using machine learning techniques. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing* (Morristown, NJ, USA, 2002), Association for Computational Linguistics, pp. 79–86.
5. Forman, G. Quantifying counts and costs via classification. *Data Min. Knowl. Discov. 17*, 2 (2008), 164–206.
6. He, B., Macdonald, C., He, J., and Ounis, I. An effective statistical approach to blog post opinion retrieval. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management* (New York, NY, USA, 2008), CIKM '08, ACM, pp. 1063–1072.
7. Huang, X., and Croft, W. B. A unified relevance model for opinion retrieval. In *CIKM '09: Proceeding of the 18th ACM conference on Information and knowledge management* (New York, NY, USA, 2009), Acm, pp. 947–956.
8. Amati, G., Amodeo, G., Capozio, V., Gaibisso, C., and Gambosi, G. On performance of topical opinion retrieval. In *SIGIR* (2010), F. Crestani, S. Marchand-Maillet, H.-H. Chen, E. N. Efthimiadis, and J. Savoy, Eds., ACM, pp. 777–778.
9. Hopkins, D., and King, G. A method of automated nonparametric content analysis for social science. *American Journal of Political Science 54*, 1 (01/2010 2010), 229–247.
10. Esuli, A., and Sebastiani, F. SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC-06, the 5th Conference on Language Resources and Evaluation* (2006).
11. Jijkoun, V., de Rijke, M., and Weerkamp, W. Generating focused topic-specific sentiment lexicons. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (Stroudsburg, PA, USA, 2010), ACL '10, Association for Computational Linguistics, pp. 585–594.
12. Amati, G., Ambrosi, E., Bianchi, M., Gaibisso, C., and Gambosi, G. Automatic construction of an opinion-term vocabulary for ad hoc retrieval. In *ECIR* (2008), C. Macdonald, I. Ounis, V. Plachouras, I. Ruthven, and R. W. White, Eds., vol. 4956 of *Lecture Notes in Computer Science*, Springer, pp. 89–100.