Elsevier SD Freedom Collection

The following terms apply ONLY to articles accessed via Elsevier SD Freedom Collection

Interlibrary Loan Notes

Follow standard ILL lending guidelines for licensed online content:
1. Make sure that the requesting library sends us a copyright compliance statement (CCG or CCL)
2. Make sure the request is not for anything other than private study, research, or scholarship.
3. Include the original copyright notice
4. No individual requests; must come from a library
5. Use secure transmission method (ArticleExchange, Odyssey, etc., but NO email)

Additional ILL restrictions:
-Can only lend to U.S. libraries and non-commercial libraries
-Each article must be checked to see if it is an open access article subject to additional terms and conditions

License Agreement Notes for Elsevier SD Freedom Collection

These terms apply as of **01/01/2014** (last updated **02/10/2017**)

The Subscriber may:
â€¢ Electronically deliver journal articles from Subscribed Titles (as defined herein) and, if any, book chapters from the Subscribed Products to fulfill requests as part of the practice commonly known as â€œinterlibrary loanâ€™â€ from non-commercial libraries located within the United States, provided that the Subscriberâ€™s staff reviews the requests and fulfills the requests in compliance with Section 108 of the U.S. Copyright Law (17 U.S.C. 108) and the Guidelines for the Proviso of Subsection 108(g)(2) (Final Report of the National Commission on new Technological Uses of Copyrighted Works, 1978). In the event that Elsevier revises its ScienceDirect â€œinterlibrary loanâ€ policy, such revised policy shall apply to the Subscriber upon notice to and written approval of the Subscriber.
Notwithstanding anything to the contrary contained in this Agreement, open access content in the Subscribed Products is subject to the terms and conditions stated in the applicable user licdnse identified in the individual journal article.

Information Sciences 367–368 (2016) 105–124

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

iSA: A fast, scalable and accurate algorithm for sentiment analysis of social media content

Andrea Ceron[a], Luigi Curini[a], Stefano Maria Iacus[b,*]

[a] Department of Social and Political Sciences, University of Milan, Milan, Italy
[b] Department of Economics, Management and Quantitative Methods, University of Milan, Milan, Italy

ARTICLE INFO

Article history:
Received 7 August 2015
Revised 10 May 2016
Accepted 29 May 2016
Available online 4 June 2016

Keywords:
Sentiment analysis
Opinion mining
Twitter analysis

ABSTRACT

We present iSA (integrated sentiment analysis), a novel algorithm designed for social networks and Web 2.0 sphere (Twitter, blogs, etc.) opinion analysis, i.e. developed for the digital environments characterized by abundance of noise compared to the amount of information. Instead of performing an individual classification and then aggregate the predicted values, iSA directly estimates the aggregated distribution of opinions. Based on supervised hand-coding rather than NLP techniques or ontological dictionaries, iSA is a language-agnostic algorithm (based on human coders' abilities). iSA exploits a dimensionality reduction approach which makes it scalable, fast, memory efficient, stable and statistically accurate. The cross-tabulation of opinions is possible with iSA thanks to its stability. Through empirical analysis it will be shown when iSA outperforms machine learning techniques of individual classification (e.g. SVM, Random Forests, etc) as well as the only other alternative for aggregated sentiment analysis known as ReadMe.

1. Introduction

The diffusion of the Internet and the striking growth of social media platforms, such as Facebook and Twitter, certainly represent one of the primary sources of the so-called "Big Data Revolution" that we are currently experiencing [12,15]. As millions of citizens begin to surf the web, create account profiles and share information online, a large amount of data becomes available. These data can then be exploited to explain and anticipate the dynamics of different events, such as stock market activity, movie success, disease outbreaks, elections, etc. [6,14,21], yielding potentially relevant results in the real world. However, the debate remains open with respect to the method that should be used to extract such information. Recognizing the relatively low informative value of merely counting the number of mentions, likes, followers and so on, the literature has largely focused on different types of sentiment analysis and opinion mining techniques [4,5,7,8].

In this paper, we present a novel, fast, scalable and accurate version of a sentiment analysis algorithm called iSA (integrated sentiment analysis), which is specifically designed for the analysis of text from the social network sphere. Instead of using an individual classification and then aggregating the estimates, iSA directly estimates the aggregated distribution of sentiment over an entire corpus of texts. The analyses presented here show that iSA outperforms other commonly used techniques with individual classification as well as the only other alternative for aggregated sentiment analysis, known as ReadMe introduced by Hopkins and King [10]. While most statistical models or text mining techniques are designed to

* Corresponding author. Tel.: +393397225246.
E-mail address: stefano.iacus@unimi.it (S.M. Iacus).

http://dx.doi.org/10.1016/j.ins.2016.05.052
0020-0255/

analyze a corpus of texts from a given and well-defined population, i.e., without misspecifications, in reality, texts from Twitter or other social networks is usually dominated by noise, no matter how accurate the data-crawling technique is. In this presentation, by "noise" or "Off-Topic" we refer to those social media posts which possibly contain the same words used to describe the relevant topics of interest to the analyst, but do not actually discuss the topics the analysis focuses on. iSA is particularly designed to work in this situation; being a supervised technique, it is completely independent of the language of the text, up to the hand coding ability of the source data, task which requires well trained native language coders in each analysis.

The paper is organized as follows. In Section 2 we introduce our theoretical framework and notation. In Section 3 we explain why the usual approach of individual classification may fail and the need of aggregate sentiment analysis estimation. Section 4 is devoted to presenting the iSA algorithm and explaining the way in which it improves on existing techniques. By focusing on the three empirical cases comprising the data set which are quite different in terms of size, content and language (English and Italian), Section 5 contrasts iSA's performances with those produced by ReadMe and two other very well-known supervised machine learning methods (the Random Forest and Support Vector Machines).

Section 6 introduces a case of sequential sampling, which is typical in most real applications, for which a modification of iSA that solves the problem and called iSAX, is described. Simulation results based on unbalanced training sets are given in Section 7. In Section 8 we introduce a simple trick to perform the cross-tabulation of data in the case of a multi-way classification using all methods. A brief discussion in Section 9 on the results concludes the paper.

2. Methods and models

Assume we have a corpus of N distinct texts. Let us denote by $\mathcal{D} = \{D_0, D_2, \ldots, D_M\}$ the set of $M+1$ possibile *categories* (i.e. sentiments or opinions) expressed in the texts. Let us denote by D_0 the most relevant category in the data in terms of the probability mass of $\{P(D), D \in \mathcal{D}\}$: the distribution of opinions in the corpus. Remark that $P(D)$ is the primary target of estimation in the content of social sciences. We reserve the symbol D_0 to the texts corresponding to Off-Topic or texts which express opinions not relevant with respect to the analysis, i.e. the *noise* in this framework. The *noise* is commonly present in any corpus of texts crawled from the social network and the Internet in general[1]. The typical workflow of analysis follows few basic steps hereafter described.

2.1. The stemming step

Once the corpus of text is available, a preprocessing step called stemming, is applied to the data. Stemming corresponds to the reduction of texts into a matrix of L stems: words, unigrams, bigrams, etc. Stop words, punctuation, white spaces, HTML code, etc., are also removed. The matrix has N rows and L columns. Let S_i, $i = 1, \ldots, K$, be a unique vector of zeroes and ones representing the presence/absence of the L possible stems. Notice that more than one text in the corpus can be represented by the same unique vector of stems S_i. The vector S_i belongs to $\mathcal{S} = \{0, 1\}^L$, the space of 0/1 vectors of length L, where each element of the vector S_i is either 1 if that stem is contained in a text, or 0 in case of absence. Thus, theoretically $K = 2^L$.

Let s_j, $j = 1, \ldots, N$, be the vector of stems associated with the individual text j in the corpus of N texts so that s_j is one and only one of the possible S_i. As $\mathcal{S}$ is potentially an incredibly large set (e.g., if $L = 10$, $2^L = 1024$ but if $L = 100$ then 2^L is in the order 10^{30}), we denote the subset of $\mathcal{S}$ with $\bar{\mathcal{S}}$, which is actually observed in a given corpus of texts, and we set $\bar{K}$ equal to the cardinality of $\bar{\mathcal{S}}$. To summarize, the relations of the different dimensions are as follows: $M \ll L < \bar{K} < N$, where "$\ll$" means "much smaller". In practice, M is usually in the order of 10 or less distinct categories, L is in the order of hundreds, $\bar{K}$ is in the order of thousands and N can be up to the millions.

2.2. The tagging step

In supervised sentiment analysis, part of the texts in the corpus, called the *training set*, is tagged (manually or according to some prescribed tool) as $d_j \in \mathcal{D}$. We assume that the subset of tagged texts is of size $n < <N$ and that there is no misclassification at this stage. The remaining set of texts of size $N - n$, for which $d_j = NA$, is called the *test set*. The whole data set is thus formalized as $\{(s_j, d_j), j = 1, \ldots, N\}$ where $s_j \in \bar{\mathcal{S}}$ and d_j can either be "NA" (not available or missing) for the training set, or one of the hand-coded categories $D \in \mathcal{D}$, for the test set. We denote the $N \times \bar{K}$ matrix of stem vectors of the whole corpus with $\Sigma = [s_j, j \in N]$. This matrix is fully observed while d_j is different from "NA" only for the training set.

2.3. The prediction (or classification) step

The typical aim of the analysis is the estimation of the aggregated distribution of opinions $\{P(D), D \in \mathcal{D}\}$ from a corpus of texts by using the individual classification of each individual text in the corpus, i.e., predict $\hat{d}_j$ from s_j, and then tabulate the distribution of $\hat{d}_j$ to estimate $P(D)$, the complete distribution of the opinions contained in the N texts.

[1] For example, in a tv political debate, any non-electoral mention to the candidates or parties are considered as D_0, or any neutral comment or news about some fact, or pure Off-Topic texts like spamming, advertising, jokes, etc.

At this step, the training set is used build a classification model (or classifier) to predict $\hat{d}_j$ from s_j, $j = 1, \ldots, N$. We denote this model as $P(D|S)$.

It is important to state that we do not assume any NLP (Natural Language Processing) rules, i.e. only stemming is applied to texts, and therefore the grammar, the order and the frequency of words is not taken into account. We work within the "bag of words" framework throughout this paper. Notice that there is a vast literature on NLP methods which is not considered in this work. In fact, as NLP assumes a well defined context of analysis, vocabulary and grammar use, etc. which is not considered here. For a review see, e.g., Aletras et al. [1], Frank and Goodman [9], Sag et al. [20].

3. Why does the traditional approach of individual classification fails in social media analysis?

At this stage we assume that the training set is randomly selected from the corpus of texts. The "traditional" approach includes all machine learning methods and statistical models that

1. use individual hand coding from the training set to construct a model $P(D|S)$ for $P(D)$ as a function of S, e.g., multinomial regression, Random Forests (RF), Support Vector Machines (SVM) etc.;
2. predict the outcome of $\hat{d}_j = D$ for texts with $S = s_j$ belonging to the test set;
3. when all the $N - n$ texts in the training set have been imputed in this way, these estimated categories $\hat{d}_j$ are aggregated, along with the exact categories d_j in the training set of n texts, to obtain a final estimate of $\hat{P}(D)$ for all N texts in the corpus.

In matrix form, we can write

$$\underset{(M+1)\times 1}{P(D)} = \underset{(M+1)\times \bar{K}}{P(D|S)} \underset{\bar{K}\times 1}{P(S)} \tag{1}$$

where $P(D)$ is a $(M+1)\times 1$ vector, $P(D|S)$ is a $(M+1)\times \bar{K}$ matrix of conditional probabilities and $P(S)$ is a $\bar{K}\times 1$ vector that represents the distribution of S_i over the corpus of texts:

$$P(D) = \begin{bmatrix} P(D = D_0) \\ P(D = D_1) \\ \vdots \\ P(D = D_M) \end{bmatrix} \qquad P(S) = \begin{bmatrix} P(S = S_1) \\ P(S = S_2) \\ \vdots \\ P(S = S_{\bar{K}}) \end{bmatrix}$$

$$P(D|S) = \begin{bmatrix} P(D = D_0|S = S_1) & \cdots & P(D = D_0|S = S_{\bar{K}}) \\ P(D = D_1|S = S_1) & \cdots & P(D = D_1|S = S_{\bar{K}}) \\ \vdots & & \\ P(D = D_M|S = S_1) & \cdots & P(D = D_M|S = S_{\bar{K}}) \end{bmatrix}$$

so that, e.g., $P(D = 0) = \sum_{j=1}^{\bar{K}} P(D = D_0|S = S_j)P(S = S_j)$, and so forth.

Once $P(D|S)$ is estimated from the training set as, say, $\hat{P}(D|S)$, then for each document in the test set with stem vector s_j, the opinion $\hat{d}_j$ is estimated with the simple Bayes estimator as the maximizer of the conditional probability, i.e., $\hat{d}_j = \arg\max_{D\in\mathcal{D}} \hat{P}(D|S = s_j)$.

In the naïve nonparametric model, the elements of the matrix $P(D|S)$, e.g., $P(D = D_i|S = S_k)$, are estimated by taking the proportion of all texts in the training set that are hand-coded as $D = D_i$, which have $s_j = S_k$ as stem vector. Any other model (SVM, etc) will do essentially the same thing in more sophisticated or more effective ways but the discussion here does not change.

Let us take $i \neq 0$, then for most S_j, these conditional probabilities are zero as the opinion D_i is expressed only for a small subset of $S_j \in \bar{\mathcal{S}}$. On the contrary, if we assume that D_0 is the category under which we confine the "*Off-Topic*" texts, i.e., texts in the corpus for which the analysis does not matter and constitute the noise, then $P(D = D_i|S = S_j) > 0$ for almost all $S_j \in \bar{\mathcal{S}}$. This means that if n is relatively small, the estimation of $P(D|S)$ will be very poor and, most of the times, the predicted category $D = d$ for a text in the test set will imputed as $d = D_0$, as D_0 is the most frequent case in the corpus of texts. As a result, $D = D_0$ will be over estimated and when the aggregation occurs, this strong bias persists so that $P(D)$ will be strongly biased as well.

Another way to understand the above issues is the following: by D_0 we denote the *noise* or "*Off-Topic*" category, but the texts belonging to D_0 are in fact coming from a different population(s) of texts, i.e. $\mathcal{D}_{noise} = \{D_0\}$, compared to the remaining $D_1, \ldots, D_M$ which are assumed to define the reference population of interest, say $\mathcal{D}_{ref} = \{D_1, \ldots, D_M\}$. In this sense, $\mathcal{D}$ is the union of two populations $\mathcal{D} = \mathcal{D}_{noise} \cup \mathcal{D}_{ref}$ and this is in fact a misspecified reference population or a noise-labeled classification problem [see e.g. 17]. Notice further that $\mathcal{D}_{noise}$ can actually consists itself of multiple heterogeneous subpopulations as $\mathcal{D}_{noise}$ is just the complementary set to $\mathcal{D}_{ref}$. Therefore, standard machine learning or any other statistical model based on $P(D|S)$ will perform inadequately. Recently new approaches have been developed [see e.g. 24] but we do not consider them in this study.

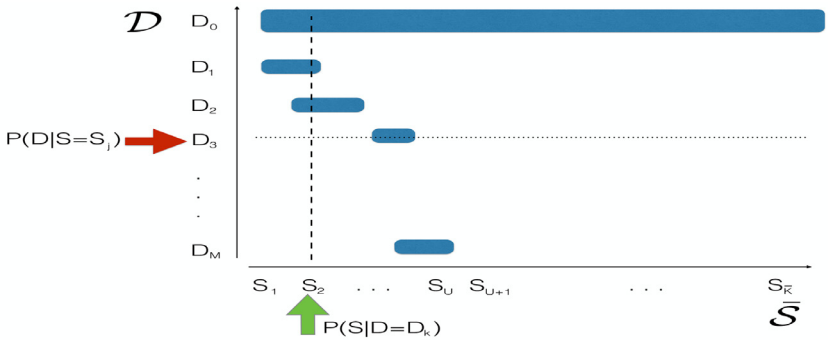

Fig. 1. The space $\bar{\mathcal{S}} \times \mathcal{D}$. The reason why, when the noise D_0 category is dominant in the data, the estimation of $P(S|D)$ is reasonably more accurate than the estimation of counterpart $P(D|S)$.

3.1. The need for aggregated estimation: reversing the point of view

Following Hopkins and King [10], the idea is to change point of view and focus on what can be accurately estimated. Instead of looking at Eq. (1), one can proceed by considering this new equation

$$\underset{\bar{K}\times 1}{P(S)} = \underset{\bar{K}\times(M+1)}{P(S|D)} \quad \underset{(M+1)\times 1}{P(D)} \tag{2}$$

where $P(S|D)$ is now a $\bar{K} \times (M+1)$ matrix of conditional probabilities whose elements $P(S = S_k|D = D_i)$ represent the frequency of a particular stem S_k given the set of texts that actually express the opinion $D = D_i$:

$$P(S|D) = \begin{bmatrix} P(S = S_1|D = D_0) & \cdots & P(S = S_1|D = D_M) \\ P(S = S_2|D = D_0) & \cdots & P(S = S_2|D = D_M) \\ \vdots & & \\ P(S = S_{\bar{K}}|D = D_0) & \cdots & P(S = S_{\bar{K}}|D = D_M) \end{bmatrix}$$

and

$$\begin{bmatrix} P(S = S_1) & = \sum_{m=0}^{M} P(S = S_1|D = D_m)P(D = D_m) \\ P(S = S_2) & = \sum_{m=0}^{M} P(S = S_2|D = D_m)P(D = D_m) \\ \vdots & \\ P(S = S_{\bar{K}}) & = \sum_{m=0}^{M} P(S = S_{\bar{K}}|D = D_m)P(D = D_m) \end{bmatrix}$$

which is Eq. (2). In this case, all of these conditional probabilities $P(S = S_j|D = D_i)$ can be considerably well estimated if there is a sufficient number of texts[2] in the training set that are hand-coded as $D = D_i$. Indeed, if $i \neq 0$, only few S_k's will be observed in texts expressing the opinion D_i. For all the remaining stems in $\bar{\mathcal{S}}$, these probabilities will be zero. On the other hand, if $i = 0$, all of these probabilities $P(S = S_j|D = D_0)$ are almost close to zero as most of the texts belong to D_0. The solution of the problem is then as follows

$$\text{(inverse problem)} \qquad \underset{(M+1)\times 1}{P(D)} = \underset{(M+1)\times(M+1)}{[P(S|D)^T P(S|D)]^{-1}} \underset{(M+1)\times\bar{K}}{P(S|D)^T} \underset{\bar{K}\times 1}{P(S)} \tag{3}$$

provided that the inverse matrix $[P(S|D)^T P(S|D)]^{-1}$ exists. Conditions for the existence of this matrix can be found in Hopkins and King [10].

Fig. 1 exemplifies the idea of aggregated estimation in formula (3). The plot represents the product space $\bar{\mathcal{S}} \times \mathcal{D}$. For simplicity, we have reordered the vectors of stems S_j on the horizontal axis in a way that, for example, S_1 to S_U, which relate to the categories D_1 to D_M, appear first and S_{U+1} to $S_{\bar{K}}$, the remaining vector of stems, appear subsequently. Clearly, the vectors S_1 to S_U sometimes also belong to D_0 and do not necessarily appear to be unique for each D_k. From this simple representation, one can see that, for example, D_2 is supported by a limited number of stems S, while D_0 is essentially supported by all stems. This makes $P(D = D_0|S_j) >> P(D = D_k|S_j)$ for all $k \neq 0$ and any given S_j (the horizontal dotted line). On the contrary, for example, $P(S = S_2|D = D_k)$ will be non-zero only for D_0, D_1 and D_2 (the vertical dashed line in the example in Fig. 1.)

[2] When this is not the case, one should increase the sample size sequentially as explained in Section 6.

Notice that Eq. (2) is very close to the regression analysis model $Y = X\beta$ and (3) to its classical least squares solution $\hat{\beta} = [X^T X]^{-1} X^T Y$ with $\beta = P(D)$, $X = P(S|D)$ and $Y = P(S)$. The only difference is that the solution (3) via linear regression analysis does not necessarily produce non-negative estimates $\beta_j \geq 0$, $j = 0, 1, \ldots, M$, such that $\sum_{j=0}^{M} \beta_j = 1$, therefore it is possible to use simple quadratic programming (QP) to solve Eq. (3).

Note that the direct solution of (3) does not allow for the individual classification of the d_j's in the training set.

3.2. The ReadMe solution to the inverse problem

The other compelling issue is that $P(S|D)$ involves the estimation over L stems, which makes a direct solution for the problem very difficult, if not impossible, to solve. For this reason Hopkins and King [10] in the ReadMe algorithm, proposed an approach based on the random sampling of the length of the stem vector, i.e., L of a given size and estimate $P(S|D)$ only for these reduced stem length vectors. For each simulation an estimate of $P(D)$ is obtained, and the results are averaged over many simulations, as is done in a statistical bagging approach. The result is a slow algorithm, in which the estimates have usually large variances. ReadMe essentially implements bagging over the explanatory variables (the stems.) Nevertheless, the ReadMe approach was, thus far, the only available and innovative method for aggregated sentiment analysis.

4. The iSA algorithm

By exploiting the ideas in Hopkins and King [10], iSA is a fast and more accurate implementation of (2) which does not require resampling method and uses the complete length of stems by a simple trick of dimensionality reduction using an idea inherited from Coarsened Exact Matching algorithm [13]. The iSA algorithm is as follows:

Step1 (collapse to one-dimensional vector). Each vector of stems, e.g. $s_j = (0, 1, 1, 0, \ldots, 0, 1)$ is transformed into a string-sequence that we denote by $C_j =$ "0110$\cdots$01"; this is the first level of dimensionality reduction of the problem: from a matrix Σ of dimension $N \times \bar{K}$ into a one-dimensional vector of length $N \times 1$.

Step2 (memory shrinking). This sequence of 0's and 1' s is further translated into hexadecimal notation such that the sequence '11110010' is recorded as $\lambda =$ 'F2' or '11100101101' as $\lambda =$ 'F2D', and so forth. So each text is actually represented by a single label λ of relatively short length. Eventually, this can be further recorded as long-integers into the memory of a computer for memory efficiency management. When step 3 is recommended (as in Section 6), the string format should be kept. Notice that, the label C_j representing the sequence s_j of, say, a hundred of 0's and 1's can be stored in just 25 characters into λ, i.e. the length is reduced to one fourth of the original one due to the hexadecimal notation.

Step2b (augmentation, optional). In the case of non-random or sequential tagging of the training set, it is recommended to split the long sequence and artificially augment the size of the corpus as follows. The sequence λ of hexadecimal codes is split into subsequences of length[3] 5, which corresponds to 30 stems in the original 0/1 representation. For example, suppose to have the sequence $\lambda_j =$ 'F2A10DEFF1AB4521A2' of 18 hexadecimal symbols and the tagged category $d_j = D_3$. The sequence λ_j is split into $4 = \lceil 18/5 \rceil$ chunks of length five or less: $\lambda_j^1 =$ "aFEA10", $\lambda_j^2 =$ "bDEFF1", $\lambda_j^3 =$ "cAB452" and $\lambda_j^4 =$ 'd1A2". At the same time, the d_j are replicated (in this example) four times, i.e. $d_j^1 = D_3$, $d_j^2 = D_3$, $d_j^3 = D_3$ and $d_j^4 = D_3$. The same applies to all sequences of the training set and those in the test set. This method results into a new data set which length is four times the original length of the data set, i.e. $4N$. When step 2b is used, we denote iSA as iSAX (where "X" stands for sample size augmentation) to simplify the exposition. The estimation of $P(\lambda|D)$ is as easy as an instant tabulation over the n labels λ's (or $4n$ after data augmentation) of the training set which has complexity of order n (or $4n$) and N (or $4N$) for $P(\lambda)$.

Step3 (quadratic programming). Whether or not step 2b has been applied, this step solves (3) into a single (QP) run. The whole set of equations becomes:

$$\begin{aligned} P(D) &= P(D|\lambda)P(\lambda) \\ P(\lambda) &= P(\lambda|D)P(D) \\ P(D) &= [P(\lambda|D)^T P(\lambda|D)]^{-1} P(\lambda|D)^T P(\lambda) \end{aligned} \tag{4}$$

provided that the inverse matrix $[P(\lambda|D)^T P(\lambda|D)]^{-1}$ exists. Remark that $P(\lambda|D)$ is more dense than the original counterpart $P(S|D)$ as there is higher probability to find similar shorter subsequences λ_j than the full sequence S_j in the data. Therefore, in iSA the original problem (3) is transformed into problem (4) and is solved directly and exactly (i.e. without simulation) using the following quadratic programming (QP):

$$\min_b \left(-\mu^T b + \frac{1}{2} b^T A b \right)$$

[3] Other length can be chosen, this does not affect the algorithm but eventually the accuracy of the estimates.

where $A = P(\lambda|D)^T P(\lambda|D)$, $\mu = P(\lambda)^T P(\lambda|D)$, $b = P(D)$ under the constraints that the coefficients are in [0, 1] and sum up to the unit.

Step4 (bootstrap, optional). In order to obtain standard errors of the point estimated $P(D)$, the rows of the original matrix Σ can be resampled according to the standard bootstrap approach and steps 1 to 3 replicated. Averaging over the replications, estimates and the empirical standard deviation can be obtained. The iSA workflow is presented in Fig. 4.

4.1. Main advantages of iSA over the state-of-the-art algorithm: ReadMe

Is it worth mentioning that the QP approach was also proposed in the case of ReadMe by Hopkins and King [10] to solve each version of the problem (3) during each replication of the bagging step. QP is not an essential part of the procedure as it can be replaced by any other algorithm to solve (3), i.e., the constrained linear regression.

The innovation element is that, in the case of iSA, the reduction of dimensionality allows for the instantaneous estimation of $P(\lambda|D)$, and therefore, a single QP step is sufficient to obtain the solution. Furthermore, it is not clear whether the resampling method proposed for ReadMe, which discards information at each step, actually guarantees a convergence of the estimates when the dimensionality of the set $\mathcal{S}$ increases too much. ReadMe is indeed theoretically unbiased in each replication, but the variability of the estimates are quite large due to bagging.

In the case of iSA, being the solution of the problem (3) direct, plain bootstrap can be used eventually to obtain correct standard errors by resampling on the observations and not the explanatory variables (the stems). Due to the fact that bagging is not used, iSA converges even when the dimensionality of $\mathcal{D}$ is very large. Indeed, iSA is generally stable.

iSA has the following main advantages over the state of the art technology ReadMe:

- The dimensionality reduction transforms the matrix Σ into a one-dimensional vector resulting in a memory efficient storage.
- The estimation of $P(S|D)$ has been replaced by the estimation of $P(\lambda|D)$ which is as easy as an instant tabulation over the n labels λ of the training set which has complexity of order n (and N for $P(\lambda)$).
- The original problem (3) is solved directly and exactly (i.e. without simulation over the subset of stems) via (4) using a single quadratic programming (QP) run.
- Computational times, as the experiments presented in the applications later on, are incredibly low compared to other methods including ReadMe.
- Correct bootstrap standard errors can be obtained with further theoretical requirements.
- It is known from empirical experience that the ReadMe algorithm might not converge or produce high variability estimates (at least the official release available in the R package ReadMe, Hopkins and King [11]) if the number of categories D grows too much. Thanks to the augmentation step and the fact that all stems are used, i.e. no bagging on the stems occurs, iSA is more stable and can work even when the size of D is quite large. This last fact allows for an additional innovation of iSA with respect to all presently available technologies: the cross tabulation of the results (see Section 8).
- Thanks again to the augmentation step, iSA works even when the size of the training set is moderately small.
- Although both iSA and ReadMe are theoretically unbiased, the uncertainty of the estimates for iSA is much lower.

To summarize, iSA is extremely fast, memory efficient, stable and accurate algorithm for which the standard errors of the estimates are validated by a standard bootstrap theory without any particular assumption or additional proof.

5. Empirical results: simple random sampling

To describe the performance of iSA, we compare this new algorithm with ReadMe, the direct competitor of aggregated sentiment analysis available in the R package `ReadMe` [11]. We also consider two other classic supervised machine learning methods: the (RF) Random Forest method [3], available in the R package `randomForest` [16], and Support Vector Machines (SVM) with a spherical kernel, as implemented in the R package `e1071` [19].

For each of the data sets presented in this section, we run a simulation experiment, taking into account only the original training set of n observations. The experiment is designed as follows: we randomly partition the n observations into two parts: $p \cdot n$ observations will constitute a new training set and $(1-p) \cdot n$ observations are considered a test set, i.e., the true category is disregarded. We let p vary between 0.25, 0.5, 0.75 and 0.9.

In this way, it is possible to evaluate the performance of each classifier. Indeed, we estimate $\hat{P}(D)$ for all texts (in the training and test sets) using iSA and ReadMe and we use the true $P(D)$ of the training set and the estimated $\hat{P}(D)$ for the test set in the case of RF and SVM. We then compare the estimated distribution using MAE (mean absolute error), i.e.

$$MAE(\text{method}) = \frac{1}{M}\sum_{i=1}^{M}\left|\hat{P}_{\text{method}}(D_i) - P(D_i)\right|$$

and the χ^2 (Chi-Squared) test

$$\chi^2(\text{method}) = \sum_{i=1}^{M}\frac{(\hat{P}_{\text{method}}(D_i) - P(D_i))^2}{P(D_i)},$$

Table 1
(Top) True distribution of $P(D)$ for the Large Movie Review data set. Fully hand-coded training set sample size $n = 25000$. (Bottom) The distribution $P(D)$ of the random sample of $n = 2500$ texts used in the simulation studies of Table 2.

Number of stars (D)	1	2	3	4	7	8	9	10	Total
target $P(D)$	20.4%	9.1%	9.7%	10.8%	10.7%	12.0%	9.1%	18.9%	100%
n. hand-coded texts	5100	2284	2420	2696	2496	3009	2263	4732	$n = 25000$
target $P(D)$	19.5%	9.2%	9.7%	11.3%	9.6%	12.0%	8.7%	20.1%	100%
n. hand-coded texts	487	229	243	282	240	299	218	502	$n = 2500$

where the "method" is one among iSA/iSAX, RF, SVM and ReadMe. The tables also report the estimation done via iSAX, a modification of iSA designed for sequential sampling, which will be introduced in Section 6. We run each experiment 100 times[4]. All computations have been performed on a Mac Book Pro, 2.7 GHz with an Intel Core i7 processor and 16 GB of RAM. All times for iSA include 100 bootstrapping replications for the standard error of the estimates even if these estimates are not shown in the Monte Carlo analysis. For the analysis we use Martin Porter's stemming algorithm and the `libstemmer` library from http://snowball.tartarus.org as implemented in the `R` package `SnowballC` [2]. After stemming, we drop the stems for which the sparsity index is greater than the q% threshold, i.e., the stems which appear less frequently than q% in the whole corpus of texts. Stop words, punctuation and white spaces are stripped from the texts as well. Two data sets consist of English texts and the additional INVALSI data set is in Italian.

5.1. The Large Movie Review data set

We start the analysis with the so-called "Large Movie Review data set" [18], which was originally designed for a different task. This data set consists of 50,000 reviews from IMDb, the *Internet Movie Database* (http://www.imdb.com), that were manually tagged as positive and negative reviews. It also includes the number of "stars" assigned by the internet users to each review. Half of these reviews are negative and half are positive. Our target $\mathcal{D}$ consists of the stars assigned to each review, a much difficult task than the dichotomous classification into positive and negative. The true target distribution of stars $P(D)$ is given in Table 1. Categories "5" and "6" do not exist in the original data base. We have $M = 8$ for this data set. The original data can be downloaded at http://ai.stanford.edu/~amaas/data/sentiment/. For the simulation experiment we confine the attention to the 25,000 observations in the original training set. In this data set, there is no miss-specification or *Off-Topic* category, so we should expect the traditional machine learning methods to perform well.

As can be seen from Table 1, the reviews are polarized and the true distribution of $P(D)$ is unbalanced: D_1 and D_{10} amount to the 40% of the total probability mass distribution, the remaining being essentially equi-distributed. Still, this case is not within the assumption of iSA or ReadMe, and one should expect a good performance from SVM and RF, given that there is no misclassification and we are performing random sampling to select the training set. The reason this happens is related to the over-representation of extreme categories (like D_1 and D_{10}) which take the role of the D_0 category.

After elementary stemming and removing stems with a sparsity index of 0.95, the remaining stems are $L = 375$, still a huge number of predictors for both SVM and RF which make their computational times very high. To reduce the computational times, we considered a random sample size of 2500 observations from the original training set of 25000.

5.1.1. Results of the simulation study

The results of the analysis are presented in Table 2. In this example, both ReadMe and iSA outperform the other methods, from small to medium training set sample sizes ($p = 25\%, 50\%$). The algorithms iSA and iSAX outperform all methods in terms of MAE and χ^2. All methods, except ReadMe[5], behave as expected as the sample size increases, i.e., the MAE, χ^2 and the Monte Carlo standard deviation of the MAE estimate (in parentheses) decrease.

Notice also that while the computational times remain essentially stable and around fraction of seconds for iSA and half a minute for ReadMe, for the other two methods, the computational times increase more than linearly with the number of observations. For example, with RF, if we pass from $p = 0.25$ to $p = 0.50$, i.e., if we double the size of the training set, the time increases by $2.76\times = 19.26/6.97$; if we move from $p = 0.25$ to $p = 0.75$, i.e., if we triple the size of the training set, the computational time increases by a factor of $4.5\times = 31.5/6.97$. To summarize, for all p's the iSA algorithm is faster, more stable and more accurate than all the other competitors. iSAX is also extremely accurate with a slightly larger variability and is still much better than ReadMe, RF and SVM, at least in this example.

5.1.2. Classification of the complete data set

This data set is the only one among the three considered completely hand-coded. We can then use all of the 25,000 observations in the original training set and the 25,000 observations of the test set to run the four classifier and compare

[4] A larger number of simulations is unfeasible in most cases given the unrealistic computational times of the methods other than iSA.

[5] This may be due to the fact that, when increasing the sample size of the training set the number of stems on which ReadMe has to perform bagging increases as well. In some cases, the algorithm does not provide stable results as the number of re-sampled stems is not sufficient and therefore, an increased number of bagging replications will be necessary. In our simulations we kept all tuning parameters fixed and we changed the sample size only.

Table 2
Monte Carlo results for the Large Movie Review data set. The table contains the MAE, Monte Carlo standard errors of MAE estimates, χ^2 statistic, and execution times for each individual replication in seconds, as multiples of the baseline, which is iSAX. Sample size $N = 2500$ observations from the original Large Movie Review training set. The number of stems is 375, threshold 95%. For the iSAX method we report the number of seconds for each individual iteration in the analysis in parentheses, which means that the total time of the simulation must be multiplied by a factor of 100. For example, while a complete analysis with iSAX for p=25% requires $0.3s \times 100 = 30s$, it requires $30s \times 26.5 = 795s = 13m25s$ and more than 1h for $p = 90\%$ for the RF algorithm.

Method	RF	SVM	ReadMe	iSA	iSAX
$p = 25\%$ ($n = 625$)					
MAE	0.093	0.151	0.012	0.009	0.016
MC std.dev.	[0.008]	[0.001]	[0.004]	[0.003]	[0.005]
χ^2	0.312	0.729	0.006	0.004	0.012
speed	(26.5x)	(6.1x)	(17.8x)	(0.2x)	(1 = 0.3 s)
$p = 50\%$ ($n = 1250$)					
MAE	0.062	0.101	0.011	0.005	0.011
MC std.dev.	[0.004]	[0.001]	[0.004]	[0.001]	[0.004]
χ^2	0.131	0.339	0.006	0.001	0.006
speed	(59.9x)	(13.5x)	(14.1x)	(0.2x)	(1 = 0.4 s)
$p = 75\%$ ($n = 1875$)					
MAE	0.031	0.050	0.014	0.003	0.008
MC std.dev.	[0.002]	[0.001]	[0.005]	[0.001]	[0.002]
χ^2	0.034	0.097	0.010	0.000	0.003
speed	(94.3x)	(25.9x)	(14.7x)	(0.2x)	(1 = 0.4 s)
$p = 90\%$ ($n = 2250$)					
MAE	0.012	0.020	0.020	0.002	0.004
MC std.dev.	[0.001]	[0.001]	[0.008]	[0.000]	[0.002]
χ^2	0.005	0.018	0.021	0.000	0.001
speed	(122.6x)	(38.1x)	(16.3x)	(0.2x)	(1 = 0.3 s)

Table 3
Classification results for the complete Large Movie Review data set. The table contains the estimated distribution of $P(D)$ for each method, the relative MAE and the computational times in seconds, relative to the classification of the set of 50,000 observations from the Large Movie Review data set where 25,000 observations are used as a training set. Number of stems 364, threshold 95%.

$n = 25000$	RF	SVM	ReadMe	iSA	iSAX
MAE	0.060	0.002	0.041	0.002	0.007
χ^2	0.120	0.000	0.116	0.000	0.002
Time	953.1 s	5289.8 s	95.2 s	2.1 s	8.0 s

the true distribution with the corresponding estimates. To this aim we disregard the hand-coding of the 25,000 observations in the test set. The results given in Table 3 show that iSA, iSAX and SVM are the most accurate methods in terms of MAE and χ^2, followed by ReadMe, and RF. Nevertheless, the iteration for iSA took only 2.1 seconds including 100 bootstrap replications while SVM (resp. RF) took 5290 (resp. 953) seconds, which is more than 2500 (resp. 462) times slower than iSA.

We will show the results for the accuracy of the estimates in Section 7.2 where we will consider confidence intervals with or without sequential sampling.

5.2. The INVALSI data set

The INVALSI ("Istituto nazionale per la valutazione del sistema educativo di istruzione e di formazione") data set is a subset of 15,398 tweets from the original data collected by *Voices from the Blogs* (a research project by the University of Milan dedicated to the monitoring of social media activity in Italy), during the days of the administration of a national test to secondary school children in Italy (from May 6th to June 18th, 2014). This written test is aimed at evaluating students' learning achievements across the country. In greater detail, the INVALSI data set includes only tweets that include the word or the hashtag `INVALSI`[6]. After stemming and removing stems with sparsity index of 0.99 we are left with $L = 149$ stems.

[6] The complete analysis, which also includes other sources of data and the complete Twitter data set can be seen in this online news post here Voices from the Blogs [23]. Unfortunately, Twitter policies do not allow for the distribution of the data, and the original IDs where lost, but the data are available upon request to the authors and upon the permission of Twitter Inc.

Table 4
True distribution of $P(D)$ for the INVALSI data set. Fully hand-coded training set sample size $n = 797$.

D	D_0	D_1	D_2	D_3	D_4	D_5	D_6
target $P(D)$	90.2%	0.9%	0.8%	3.4%	0.8%	2.3%	1.8%
n. hand-coded texts	719	7	6	27	6	18	14

Table 5
Monte Carlo results for the INVALSI data set. The table contains the MAE, Monte Carlo standard errors of MAE estimates, χ^2 statistic, and execution times for each individual replication in seconds as multiples of the baseline, which is iSAX. Number of stems: 149, threshold 99%. $N = 797$.

Method	RF	SVM	ReadMe	iSA	iSAX
$p = 25\%$ ($n = 199$)					
MAE	0.020	0.015	0.023	0.010	0.014
MC std.dev.	[0.002]	[0.003]	[0.010]	[0.004]	[0.007]
χ^2	0.046	0.030	0.042	0.016	0.031
speed	(35.8x)	(3.0x)	(259.4x)	(1.9x)	(1 = 0.02 s)
$p = 50\%$ ($n = 398$)					
MAE	0.013	0.011	0.026	0.006	0.011
MC std.dev.	[0.002]	[0.002]	[0.010]	[0.002]	[0.005]
χ^2	0.018	0.014	0.048	0.007	0.023
speed	(84.3x)	(8.2x)	(118.6x)	(1.5x)	(1 = 0.02 s)
$p = 75\%$ ($n = 598$)					
MAE	0.007	0.006	0.041	0.003	0.007
MC std.dev.	[0.001]	[0.001]	[0.013]	[0.001]	[0.001]
χ^2	0.005	0.004	0.089	0.003	0.010
speed	(144.3x)	(23.1x)	(103.4x)	(1.4x)	(1 = 0.02 s)
$p = 90\%$ ($n = 717$)					
MAE	0.003	0.002	0.064	0.002	0.004
MC std.dev.	[0.001]	[0.001]	[0.018]	[0.001]	[0.002]
χ^2	0.001	0.001	0.173	0.001	0.005
speed	(171.7x)	(32.5x)	(95.7x)	(1.3x)	(1 = 0.02s)

The opinions where hand-coded as D_0 = "Off Topic" (or not expressing an opinion, just stating news about the event) and then from D_1 to D_6 with meaningful topics, therefore $M = 7$ for this data set[7]. In Voices from the Blogs [23] the data have been aggregated differently, but here this information is not relevant for the ongoing simulation study. The target distribution is given in Table 4.

As is clear from Table 4, this is an extremely difficult case to handle as most of the texts fall into the D_0 category and the signal is very faint compared to the noise. The training set is composed of 797 hand-coded texts, and we perform the same simulation experiment of Section 5.1. The results of the analysis are presented in Table 5.

As one can see from Table 5, the ReadMe algorithm tends to increase the MAE and variability of the estimates as the sample size increases. This is because the number of coded texts for D_1, D_2 and D_4 is extremely small; therefore, as the sample size increases, the information about the three categories remains essentially fixed while the information for the other categories increases. An empirical analysis suggests that, on average, at least 20 hand-coded texts for each category are necessary to stabilize the estimates for ReadMe. However, this is not a issue for the iSA and iSAX methods, as we will discuss in Section 6. To summarize, also in this case, iSA and iSAX are faster, more stable and more accurate than all other competitors. RF and SVM behave similarly to iSA for $p = 90\%$.

5.3. The Expo2015 data set

This data set consists of a subset of 28,195 tweets concerning international reactions (i.e. excluding Italy) on the forthcoming Expo2015 event (see http://www.expo2015.org/en/index.html) in Milan. The tweets[8] were written in English and published worldwide between the 1st of November 2013 and the 31st of May 2014, using the keyword expo2015. After stemming and removing the stems with a sparsity index of 0.99, we are left with $L = 578$ stems. The opinions were hand-coded as D_0 = "Off Topic" (or not expressing an opinion, just news about the event) and then from D_1 to D_{10} with

[7] Although this is irrelevant to the reader, the 7 topics where: D_1 = "boycott", D_2 = "unfair", D_3 = "useless", D_4 = "meritocratic", D_5 = "modern", D_6 = "notional".

[8] These data have also been collected by Voices from the Blogs and are available upon request to the authors.

Table 6
True distribution of $P(D)$ for the Expo2015 data set. Fully hand-coded training set sample size $n = 597$.

D	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}
target $P(D)$	48.2%	4.0%	4.0%	2.8%	3.9%	4.4%	8.7%	3.5%	0.7%	4.2%	15.6%
n. hand-coded texts	288	24	24	17	23	26	52	21	4	25	93

Table 7
Monte Carlo results for the Expo2015 data set. The table contains the MAE, Monte Carlo standard errors, χ^2, and execution times for each individual replication in seconds as multiples of the baseline, which is in the iSAX column. Number of stems: 578, threshold 99%. $N = 597$.

Method	RF	SVM	ReadMe	iSA	iSAX
$p = 25\%$ ($n = 149$)					
MAE	0.049	0.039	0.031	0.016	0.020
MC std.dev.	[0.006]	[0.008]	[0.012]	[0.004]	[0.006]
χ^2	0.192	0.130	0.110	0.033	0.057
speed	(52.4x)	(3.6x)	(232.2x)	(0.6x)	(1 = 0.06 s)
$p = 50\%$ ($n = 298$)					
MAE	0.029	0.025	0.015	0.010	0.015
MC std.dev.	[0.004]	[0.005]	[0.010]	[0.003]	[0.005]
χ^2	0.067	0.052	0.089	0.013	0.032
speed	(105.6x)	(4.1x)	(72.2x)	(0.5x)	(1 = 0.06 s)
$p = 75\%$ ($n = 448$)					
MAE	0.014	0.012	0.040	0.006	0.010
MC std.dev.	[0.002]	[0.002]	[0.010]	[0.002]	[0.003]
χ^2	0.016	0.012	0.136	0.005	0.014
speed	(165.6x)	(4.8x)	(63.4x)	(0.5x)	(1 = 0.06 s)
$p = 90\%$ ($n = 537$)					
MAE	0.006	0.005	0.062	0.003	0.006
MC std.dev.	[0.001]	[0.001]	[0.016]	[0.001]	[0.002]
χ^2	0.003	0.002	0.311	0.002	0.005
speed	(199.3x)	(5.2x)	(55.5x)	(0.5x)	(1 = 0.07 s)

meaningful topics[9]. Notice that D_8 has only 4 coded texts in the training sample. Again, the real content of the topic is not relevant in this simulation study but the extended analysis can be seen in Voices from the Blogs [22]. The target distribution is given in Table 6 and the results of the Monte Carlo analysis are given in Table 7.

In this example (see Table 7), ReadMe tends to be very unstable for two reasons: D_8 has too few hand coded texts (as in the INVALSI data set), and the number of categories D is quite large, $M = 11$ in this case. Again, all methods except ReadMe, present correct asymptotic behavior. To summarize, also in this example, iSA and iSAX are the fastest, most accurate and stable algorithms.

6. The iSAX algorithm for sequential sampling

The dimensionality reduction used by iSA has the effect of assuming a reduced variability in the configuration of stems, which means that if a vector of stems in the test set is not observed in the training set, the corresponding entry of $P(S|D)$ will be estimated as zero. This is not a problem when the training set is sufficiently large, i.e., n is large and randomly selected from the entire corpus of texts due to the law of large numbers.

However, in operational setups the tagging of texts occurs in a sequential way, i.e., the coders usually skip many texts while looking for those texts which contain the relevant categories D or a filter is applied by the researcher to a random set of texts before sending out the texts for manual tagging. This results in a sequential procedure that gives a training set which is not obtained by simple random sampling. In this situation, most machine learning methods of individual classification will fail in the presence of training sets with small sample sizes due to the mismatch of representativity of the training set with respect to the test set. This is the second reason why the ReadMe approach randomly samples the set of stems in an attempt to get a better estimate of $P(D)$. Indeed, to avoid bias in the estimates and increase accuracy, a sufficient number of hand-coded texts is needed for each opinion $D \in \mathcal{D}$. The empirical evidence seems to require at least 20 tagged texts per category D.

On the contrary, it is still possible to attain the same goal in the presence of sequential sampling by modifying the iSA algorithm as follows. The sequence ℓ of the hexadecimal codes is split into subsequences of length 5, which corresponds to 20 stems in the original 0/1 representation. For example, suppose to have the sequence ℓ_j = 'F2A10DEFF1AB4521A2' of 18

[9] D_1 = "architecture", D_2 = "business", D_3 = "cooperation", D_4 = "design", D_5 = "environment & food", D_6 = "events & mascotte", D_7 = "innovation", D_8 = "sustainability", D_9 = "tourism", D_{10} = "worldwide participation".

Table 8
Monte Carlo results for the Large Movie Review data set. The table contains the MAE, Monte Carlo standard errors of MAE estimates, χ^2 test statistic, and execution times for each individual replication in seconds as multiples of the baseline, which is iSAX. The training set is made by sampling n hand-coded texts for each of the $M = 8$ categories D to break the proportionality. The total number of observations is $N = 5000$ sampled from the original Large Movie Review data set. Number of stems 367, threshold 95%.

Method	RF	SVM	ReadMe	iSA	iSAX
$n = 10M = 80$ (1.6%)					
MAE	0.043	0.107	0.034	0.036	0.036
MC std.dev.	[0.013]	[0.040]	[0.005]	[0.002]	[0.004]
χ^2	0.091	0.496	0.048	0.051	0.053
speed	(2.2x)	(0.5x)	(10.0x)	(0.1x)	(1 = 0.9 s)
$n = 25M = 200$ (4.0%)					
MAE	0.028	0.168	0.032	0.035	0.034
MC std.dev.	[0.009]	[0.012]	[0.004]	[0.001]	[0.005]
χ^2	0.041	0.981	0.043	0.050	0.048
speed	(3.5x)	(1.0x)	(10.1x)	(0.1x)	(1 = 1.1 s)
$n = 50M = 400$ (8.0%)					
MAE	0.022	0.175	0.030	0.036	0.033
MC std.dev.	[0.006]	[0.009]	[0.004]	[0.001]	[0.005]
χ^2	0.025	1.019	0.039	0.050	0.046
speed	(5.6x)	(1.5x)	(8.9x)	(0.1x)	(1 = 1.2 s)
$n = 100M = 800$ (16.0%)					
MAE	0.019	0.167	0.028	0.036	0.031
MC std.dev.	[0.005]	[0.008]	[0.004]	[0.000]	[0.005]
χ^2	0.019	0.920	0.033	0.050	0.041
speed	(14.0x)	(4.3x)	(10.8x)	(0.2x)	(1 = 1.2 s)
$n = 300M = 2400$ (48.0%)					
MAE	0.016	0.110	0.021	0.036	0.026
MC std.dev.	[0.002]	[0.004]	[0.004]	[0.000]	[0.004]
χ^2	0.012	0.467	0.023	0.050	0.030
speed	(41.4x)	(15.9x)	(9.3x)	(0.1x)	(1 = 1.0 s)

Table 9
Classification results for the complete Large Movie Review data set. The table contains the estimated distribution of $P(D)$ for each method, the relative MAE and the computational times in seconds relative to the classification of the set of 50,000 observations from the Large Movie Review data set, where 25,000 observations are used as training set (top) and (bottom) and where only 10 observations per category have been chosen for the training set (sample size: training set = 80, test set = 49840). A total of 100 bootstrap replications for the evaluation of the standard errors of iSA and iSAX estimates are used. Number of stems 364, threshold 95%.

$n = 25000$	RF	SVM	ReadMe	iSA	iSAX
MAE	0.060	0.002	0.040	0.002	0.007
χ^2	0.120	0.000	0.116	0.000	0.010
Time	953.1s	5289.8s	95.2s	2.1s	8.0s
$n = 80$	RF	SVM	ReadMe	iSA	iSAX
MAE	0.055	0.037	0.037	0.037	0.039
χ^2	0.135	0.052	0.054	0.055	0.062
Time	20.4s	0.6s	115.3s	2.9s	10.4s

hexadecimal symbols and the tagged category $d_j = D_3$. The sequence ℓ_j is split into $4 = \lceil 18/5 \rceil$ chunks of a length of five or less: ℓ_j^1 = 'aF2A10', ℓ_j^2 = 'bDEFF1', ℓ_j^3 = 'cAB451' and ℓ_j^4 = 'eA2'. At the same time, the d_j are replicated (in this example) four times, i.e., $d_j^1 = D_3$, $d_j^2 = D_3$, $d_j^3 = D_3$ and $d_j^4 = D_3$. The same applies to all sequences of the training set and those in the test set. This method results in a new data set of length that is four times the original length of the data set, i.e., $4N$. For this new data set, the standard iSA can be applied and the problem (3) is solved in one iteration of QP with the same complexity as before. Clearly, in this approach there is no need to average the estimates like in ReadMe because there will only be one solution. Again, bootstrapping on the vector of stems can be used to obtain the standard errors of the new iSA estimates.

Table 10
Classification results for the complete Large Movie Review data set. Data are as in Table 9 for the whole data set of 50000 observations with $n = 25000$. Up: the final estimated distributions, Bottom: the 95% confidence interval upper-bound and lower-bound estimates for iSA and iSAX.

Stars	True	iSAX	ReadMe	RF	SVM	iSA
1	0.202	0.204	0.201	0.318	0.204	0.204
2	0.092	0.091	0.241	0.046	0.091	0.091
3	0.099	0.097	0.111	0.052	0.097	0.097
4	0.107	0.108	0.099	0.070	0.108	0.108
7	0.096	0.100	0.098	0.061	0.100	0.100
8	0.117	0.121	0.076	0.089	0.120	0.121
9	0.092	0.090	0.094	0.046	0.091	0.090
10	0.195	0.189	0.080	0.318	0.189	0.189
MAE		0.002	0.041	0.060	0.002	0.002
χ^2		0.000	0.125	0.120	0.000	0.000

Stars	Lower	True	iSA	Upper	Stars	Lower	True	iSAX	Upper
1	0.202	0.202	0.204	0.206	1	0.188	0.202	0.204	0.200
2	0.190	0.092	0.091	0.093	2	0.083	0.092	0.091	0.093
3	0.096	0.099	0.097	0.099	3	0.088	0.099	0.097	0.101
4	0.106	0.107	0.108	0.109	4	0.092	0.107	0.108	0.105
7	0.098	0.096	0.100	0.101	7	0.076	0.096	0.100	0.086
8	0.119	0.117	0.121	0.122	8	0.100	0.117	0.121	0.111
9	0.089	0.092	0.090	0.092	9	0.077	0.092	0.090	0.085
10	0.187	0.195	0.189	0.191	10	0.210	0.195	0.189	0.218

Table 11
Classification results for the complete Large Movie Review data set. Data are as in Table 9 for the whole data set of 50000 observations with $n = 80$. Up: the final estimated distributions, Bottom: the 95% confidence interval upper-bound and lower-bound estimates for iSA and iSAX.

Stars	True	iSAX	ReadMe	RF	SVM	iSA
1	0.202	0.110	0.117	0.125	0.125	0.120
2	0.092	0.132	0.115	0.141	0.125	0.126
3	0.099	0.141	0.125	0.090	0.125	0.127
4	0.107	0.120	0.132	0.096	0.125	0.122
7	0.096	0.127	0.125	0.097	0.125	0.130
8	0.117	0.122	0.121	0.194	0.125	0.123
9	0.092	0.116	0.131	0.185	0.125	0.124
10	0.195	0.132	0.134	0.073	0.125	0.128
MAE		0.039	0.037	0.055	0.037	0.037
χ^2		0.062	0.054	0.135	0.052	0.055

Stars	Lower	True	iSA	Upper	Stars	Lower	True	iSAX	Upper
1	0.060	0.202	0.120	0.179	1	0.087	0.202	0.110	0.132
2	0.055	0.092	0.126	0.197	2	0.111	0.092	0.132	0.153
3	0.059	0.099	0.127	0.194	3	0.118	0.099	0.141	0.165
4	0.057	0.107	0.122	0.186	4	0.099	0.107	0.120	0.141
7	0.060	0.096	0.130	0.200	7	0.108	0.096	0.127	0.145
8	0.043	0.117	0.123	0.204	8	0.096	0.117	0.122	0.148
9	0.055	0.092	0.124	0.193	9	0.094	0.092	0.116	0.239
10	0.059	0.195	0.128	0.197	10	0.108	0.195	0.132	0.156

Notice that this new version of iSA works almost equally well in the case of a truly random training set and in the general applied framework of sequential coding. This version of iSA has larger variability than the version presented in Section 4. To distinguish the two versions of iSA, we denote the latter as iSAX.

7. Empirical results: sequential sampling

In this experiment we create a random sample from each data set which contains the same number of entries per category D. This is to mimic the case of sequential random sampling, although only approximately as this sample is still random. This type of sampling approximates the case where the distribution of $P(D)$ in training set is quite different to the target distribution.

Table 12
Monte Carlo results for the INVALSI data set. The table contains the MAE, Monte Carlo standard errors, χ^2, and execution times for each individual replication in seconds as multiples of the baseline, which is in the iSAX column. The training set is made by sampling n hand-coded texts per each category D to broke the proportionality. Total number of observations 797. Number of stems 149, threshold 99%.

Method	RF	SVM	ReadMe	iSA	iSAX
$n = 5$ (4.4%)					
MAE	0.214	0.211	0.220	0.215	0.218
MC std.dev.	[0.038]	[0.024]	[0.016]	[0.004]	[0.012]
χ^2	1.190	1.128	1.209	1.144	1.183
speed	(7.8x)	(2.7x)	(103.0x)	(1.3x)	(1 = 0.02 s)
$n = 10$ (7.4%)					
MAE	0.199	0.192	0.210	0.206	0.212
MC std.dev.	[0.031]	[0.016]	[0.014]	[0.005]	[0.013]
χ^2	1.031	0.956	0.054	0.050	0.049
speed	(12.0x)	(2.6x)	(92.8x)	(1.0x)	(1 = 0.02 s)
$n = 15$ (9.8%)					
MAE	0.186	0.182	0.201	0.200	0.202
MC std.dev.	[0.029]	[0.014]	[0.012]	[0.005]	[0.011]
χ^2	0.904	0.848	1.014	1.000	1.019
speed	(16.9x)	(2.9x)	(99.9x)	(1.1x)	(1 = 0.02 s)

Table 13
Monte Carlo results for the Expo2015 data set. The table contains the MAE, Monte Carlo standard errors, χ^2, and execution times for each individual replication in seconds as multiples of the baseline, which is in the iSAX column. The training set is made by sampling n hand-coded texts per each category D to broke proportionality. The total number of observations is 597. Number of stems 578, threshold 99%.

Method	RF	SVM	ReadMe	iSA	iSAX
$n = 5$ (9%)					
MAE	0.092	0.090	-	0.082	0.083
MC std.dev.	[0.027]	[0.016]	-	[0.004]	[0.010]
χ^2	1.190	1.128	-	1.144	1.183
speed	(27.2x)	(3.4x)	-	(0.5x)	(1 = 0.06 s)
$n = 10$ (17.4%)					
MAE	0.086	0.079	0.083	0.080	0.078
MC std.dev.	[0.021]	[0.011]	[0.009]	[0.004]	[0.010]
χ^2	0.581	0.492	0.054	0.459	0.452
speed	(38.1x)	(3.5x)	(181.1x)	(0.4x)	(1 = 0.06 s)
$n = 20$ (33.6%)					
MAE	0.083	0.077	0.078	0.077	0.077
MC std.dev.	[0.014]	[0.009]	[0.008]	[0.002]	[0.009]
χ^2	0.534	0.448	0.443	0.417	0.426
speed	(62.4x)	(3.6x)	(47.3x)	(0.4x)	(1 = 0.06 s)

7.1. Analysis of the Large Movie Review data

We let the number of observations in the training set for each category D vary in the set {10, 25, 50, 100, 300}. In real applications, the number of hand-coded texts is generally approximately 20. Looking at the results in Table 8 one can see that iSA and iSAX are quite efficient for small sample sizes while for the unrealistic case of 300 hand-coded texts per category, RF seems to be appreciably better than the other methods although all methods provide very similar MAE and χ^2 statistics, apart from SVM. We also tried to use a very small sample size to predict all 50,000 original entries in the Movie Review Database and compare it with a training set with a size of 25,000. Table 9 shows that iSA and iSAX are very powerful in both situations and dominate the other methods in terms of MAE and χ^2. While the computing time for machine learning methods (RF and SVM) depends on the number of observations in the training set, for iSA, iSAX and ReadMe the timing depends on the entire size of the data. In addition, for ReadMe, the timing also depends on the number of categories D and the number of items coded per category.

Table 14
The Renzi data set. The table contains the two-ways table $D^{(1)}$ against $D^{(2)}$ (Up) and the recoded distribution $D = D^{(1)} \times D^{(2)}$ (Bottom) that is used to run the analysis. The training set consists of $n = 1324$ hand-coded texts. The total number of texts in the corpus $N = 39845$. Number of stems 216, threshold 95%.

$D^{(1)} \times D^{(2)}$	C01: Negative	C02: Neutral	C03: Positive	C04: Off-Topic	Total
R01: Environment	10		45		55
R02: Electoral campaign	60	3	4		67
R03: Economy	80	2	5		87
R04: Europe	11				11
R05: Law & Justice	54	3	30		87
R06: Immigration & Homeland security	48	4	6		58
R07: Work	23	1	4		28
R08: Electoral Reform	46	5	5		56
R09: School	445	46	79		570
R10: Off-Topic				305	305
Total	777	64	178	305	1324

D	R01-C01	R01-C03	R02-C01	R02-C02	R02-C03	R03-C01	R03-C02	R03-C03	R04-C01	R05-C01
count	10	45	60	3	4	80	2	5	11	54
D	R05-C02	R05-C03	R06-C01	R06-C02	R06-C03	R07-C01	R07-C02	R07-C03	R08-C01	R08-C02
count	3	30	48	4	6	23	1	4	46	5
D	R08-C03	R09-C01	R09-C02	R09-C03	R10-C03	Total				
count	5	445	46	79	305	1324				

Table 15
The Renzi data set. The estimated joint distribution of $D^{(1)}$ against $D^{(2)}$ (Top), conditional distribution of $D^{(2)}|D^{(1)}$ (Middle) and conditional distribution of $D^{(1)}|D^{(2)}$ (Bottom) using iSAX. All data are as in Table 14.

	Negative	Neutral	Positive	Off-Topic	Total
Environment	1.54%		2.07%		3.61%
Electoral campaign	6.06%	0.64%	0.79%		7.48%
Economy	6.70%	0.37%	1.15%		8.23%
Europe	1.35%				1.35%
Law & Justice	6.35%	0.67%	2.20%		9.22%
Immigration & Homeland security	6.82%	1.19%	1.03%		9.05%
Work	1.75%	0.13%	1.03%		2.91%
Electoral Reform	3.31%	1.11%	0.95%		5.37%
School	19.42%	1.13%	3.54%		24.08%
Off-Topic				28.70%	28.70%
Total	53.30%	5.24%	12.76%	28.70%	100%

Conditional distribution $D^{(2)}\|D^{(1)}$	Negative	Neutral	Positive	Off-Topic	Total
Environment	42.65%		57.35%		100.00%
Electoral campaign	80.96%	8.52%	10.52%		100.00%
Economy	81.48%	4.49%	14.03%		100.00%
Europe	100.00%				100.00%
Law & Justice	68.83%	7.29%	23.89%		100.00%
Immigration & Homeland security	75.43%	13.17%	11.40%		100.00%
Work	60.10%	4.60%	35.30%		100.00%
Electoral Reform	61.66%	20.68%	17.66%		100.00%
School	80.62%	4.68%	14.70%		100.00%
Off-Topic				100.00%	100.00%

Conditional distribution $D^{(1)}\|D^{(2)}$	Negative	Neutral	Positive	Off-Topic
Environment	2.88%		16.20%	
Electoral campaign	11.37%	12.16%	6.17%	
Economy	12.58%	7.05%	9.05%	
Europe	2.54%			
Law & Justice	11.91%	12.82%	17.26%	
Immigration & Homeland security	12.80%	22.73%	8.08%	
Work	3.29%	2.55%	8.06%	
Electoral Reform	6.21%	21.17%	7.43%	
School	36.43%	21.51%	27.74%	
Off-Topic				100.00%
Total	100.00%	100.00%	100.00%	100.00%

Table 16
The Renzi data set. The estimated joint distribution of $D^{(1)}$ against $D^{(2)}$ with iSAX, RF, ReadMe and SVM on a sample of 111 texts from the training set of 1324. The top table corresponds to the joint distribution in the training set of 1324 hand-coded texts. All other data are as in Table 14.

True	Negative	Neutral	Positive	Off-Topic	Total
Environment	0.76%		3.40%		4.15%
Electoral campaign	4.53%	0.23%	0.30%		5.06%
Economy	6.04%	0.15%	0.38%		6.57%
Europe	0.83%				0.83%
Law & justice	4.08%	0.23%	2.27%		6.57%
Immigration & homeland security	3.63%	0.30%	0.45%		4.38%
Work	1.74%	0.08%	0.30%		2.11%
Electoral reform	3.47%	0.38%	0.38%		4.23%
School	33.61%	3.47%	5.97%		43.05%
Off-Topic				23.04%	23.04%
Total	58.69%	4.83%	13.44%	23.04%	100%
iSAX	Negative	Neutral	Positive	Off-Topic	Total
Environment	3.53%		4.77%		8.30%
Electoral campaign	4.95%	1.56%	2.14%		8.65%
Economy	4.84%	0.94%	2.65%		8.43%
Europe	4.01%				4.01%
Law & justice	5.38%	1.58%	4.02%		10.97%
Immigration & homeland security	5.87%	2.83%	3.09%		11.79%
Work	5.28%	0.33%	2.16%		7.77%
Electoral reform	5.43%	2.56%	2.66%		10.65%
School	8.19%	9.01%	6.39%		23.59%
Off-Topic				5.84%	5.84%
Total	47.49%	18.81%	27.86%	5.84%	100%
MAE = 0.035, $\chi^2 = 0.504$					
RF	Negative	Neutral	Positive	Off-Topic	Total
Environment	8.01%		2.11%		10.12%
Electoral campaign	3.47%	0.30%	0.45%		4.23%
Economy	3.85%	0.23%	0.38%		4.46%
Europe	2.11%				2.11%
Law & justice	4.53%	0.45%	1.06%		6.04%
Immigration & homeland security	1.96%	0.30%	0.98%		3.25%
Work	17.90%	0.08%	0.60%		18.58%
Electoral reform	3.78%	0.76%	0.76%		5.29%
School	20.09%	10.57%	14.12%		44.79%
Off-Topic				1.13%	1.13%
Total	65.71%	12.69%	20.47%	1.13%	100%
MAE = 0.034, $\chi^2 = 0.526$					
ReadMe	Negative	Neutral	Positive	Off-Topic	Total
Environment	2.14%		2.30%		4.44%
Electoral campaign	2.45%	1.27%	4.92%		8.64%
Economy	4.55%	1.78%	2.10%		8.43%
Europe	4.47%				4.47%
Law & justice	3.15%	1.26%	2.30%		6.71%
Immigration & homeland security	6.90%	1.83%	2.08%		10.81%
Work	2.74%	1.63%	2.33%		6.70%
Electoral reform	5.47%	2.65%	1.63%		9.75%
School	12.13%	12.96%	10.31%		35.40%
Off-Topic				4.66%	4.66%
Total	44.00%	23.38%	27.96%	4.66%	100%
MAE = 0.036, $\chi^2 = 0.506$					
SVM	Negative	Neutral	Positive	Off-Topic	Total
Environment	5.49%		5.49%		10.99%
Electoral campaign	5.49%	1.65%	2.20%		9.34%
Economy	5.49%	1.10%	2.75%		9.34%
Europe	5.49%				5.49%
Law & justice	5.49%	2.20%	3.30%		10.99%
Immigration & homeland security	5.49%	2.20%	3.30%		10.99%
Work	5.49%	0.55%	2.20%		8.24%
Electoral reform	5.49%	2.75%	2.75%		10.99%
School	5.49%	5.49%	5.49%		16.48%
Off-Topic				5.49%	5.49%
Total	49.45%	15.38%	29.67%	5.49%	100%
MAE = 0.037, $\chi^2 = 0.586$					

Table 17
The Renzi data set. The estimated joint distribution of $D^{(1)}$ against $D^{(2)}$ with RF (top), ReadMe (middle) and SVM (bottom). All data are as in Table 14.

RF	Negative	Neutral	Positive	Off-Topic	Total
Environment	0.03%		2.61%		2.63%
Electoral campaign	1.40%	0.01%	0.01%		1.42%
Economy	8.05%	0.01%	0.02%		8.07%
Europe	0.05%				0.05%
Law & justice	2.76%	0.04%	0.10%		2.89%
Immigration & homeland security	2.27%	0.01%	0.03%		2.31%
Work	0.25%	0.01%	0.23%		0.48%
Electoral reform	4.17%	0.01%	0.02%		4.20%
School	32.10%	0.17%	1.02%		33.29%
Off-Topic				44.65%	44.65%
Total	51.07%	0.26%	4.02%	44.65%	100%
ReadMe	Negative	Neutral	Positive	Off-Topic	Total
Environment	0.56%		4.62%		5.18%
Electoral campaign	6.22%	2.00%	4.40%		12.61%
Economy	13.60%	2.54%	2.68%		18.82%
Europe	4.84%				4.84%
Law & justice	4.14%	1.61%	4.28%		10.03%
Immigration & homeland security	1.60%	1.85%	3.23%		6.67%
Work	1.28%	1.09%	1.68%		4.05%
Electoral reform	5.51%	2.94%	3.09%		11.54%
School	9.46%	2.46%	3.44%		15.35%
Off-Topic				10.91%	10.91%
Total	47.20%	14.48%	27.41%	10.91%	100%
SVM	Negative	Neutral	Positive	Off-Topic	Total
Environment	0.76%		3.40%		4.15%
Electoral campaign	4.53%	0.23%	0.30%		5.06%
Economy	6.04%	0.15%	0.38%		6.57%
Europe	0.83%				0.83%
Law & justice	4.08%	0.23%	2.27%		6.57%
Immigration & homeland security	3.63%	0.30%	0.45%		4.38%
Work	1.74%	0.08%	0.30%		2.11%
Electoral reform	3.47%	0.38%	0.38%		4.23%
School	33.61%	3.47%	5.97%		43.05%
Off-Topic				23.04%	23.04%
Total	58.69%	4.83%	13.44%	23.04%	100%

7.2. Confidence intervals

We finally evaluate the 95% confidence intervals for iSA and iSAX in both cases in Tables 10 and 11. The other methods require further bootstrap analyses in order to produce standard errors, which make the experiment unfeasible, so we did not consider standard errors for ReadMe, SVM and RF. In Tables 10 and 11, we can see that in all cases, both the iSA and iSAX confidence intervals contain the true values of the parameters. The only cases in which the true value is outside the confidence interval are for the extreme categories D_1 = "1 Star" and D_8 = "10 Stars", which is, as seen in all other tables, an extremely difficult case for all methods.

7.3. Analysis for the INVALSI data

In the case of the INVALSI data set, we let the number of observations in the training set for each category D to vary in the set {5, 10, 15}. When the number of actual hand-coded text is less then this threshold, we take all the hand-coded texts available. In this example, all methods provide similar results as reported in Table 12 although SVM is slightly better in few cases.

7.4. Analysis for the Expo2015 data

Finally, for the Expo2015 data set the iSA and iSAX dominate all methods although all methods provide quite similar values in term of MAE and χ^2 as shown in Table 13. Notice that for very small sample sizes ReadMe could not find a solution.

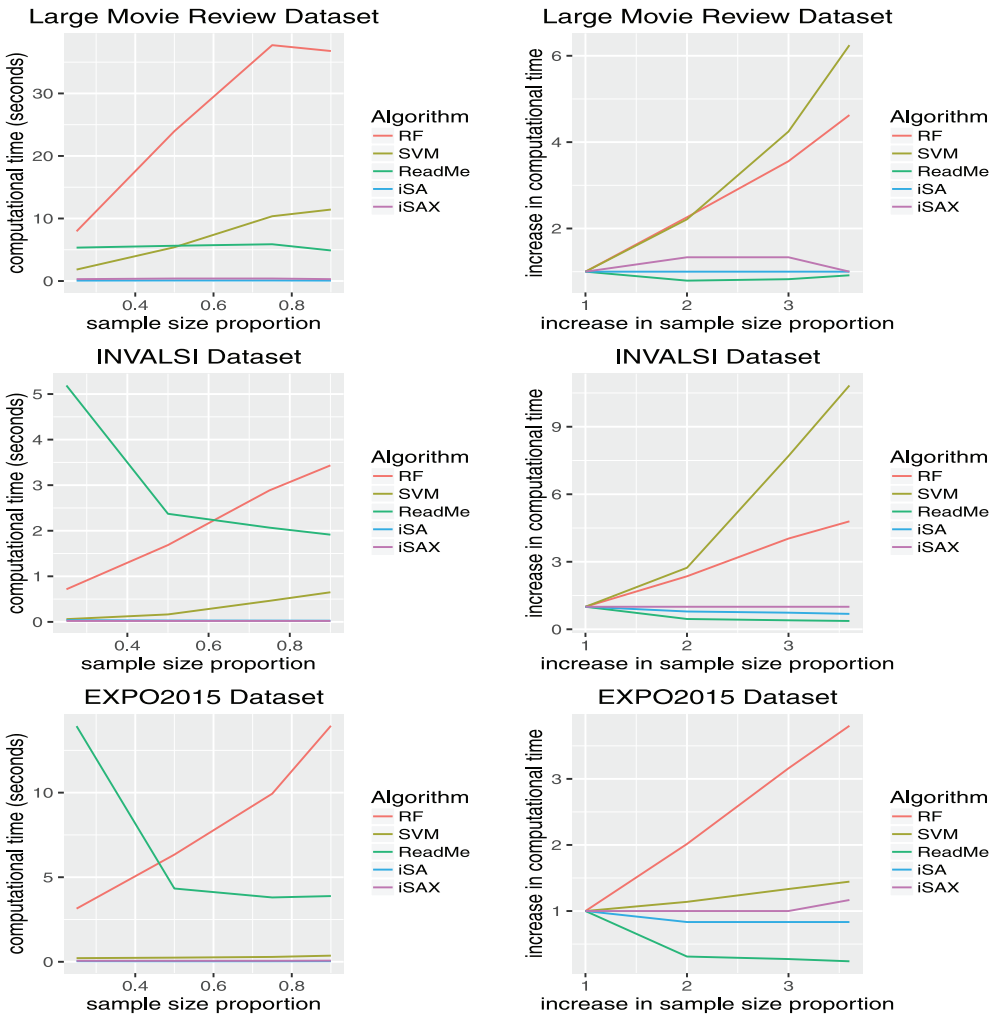

Fig. 2. Computational efficiency. Left panel: (absolute) computational time in seconds for each iteration of the MC analysis. Right panel: (relative) computational times growth as a function of the sample size growth.

8. Cross-tabulation

The ability of iSA/iSAX to work even when the sample size of the training set is very small can be exploited to run a cross-tabulation of categorization when a corpus of texts is hand-coded/tagged along multiple dimensions. Suppose there is a training set where $D^{(1)}$ is the tagging for the first dimension on $M^{(1)}$ possible values and $D^{(2)}$ is the tag for the second dimension on $M^{(2)}$ possible values; $M^{(1)}$ is not necessarily the same as $M^{(2)}$. We can consider the cross-product of the values $D^{(1)} \times D^{(2)} = D$ so that D will have $M = M^{(1)} \cdot M^{(2)}$ possible distinct values, not all of which are available in the training set. We can now apply iSA/iSAX to this new tag variable D, estimate $P(D)$ as in the above, and then reconstruct the bivariate distribution ex-post. To show this capability we show an application based on a different data set: the Renzi data set.

8.1. The Renzi data set

This data set consists of a corpus of $N = 39845$ text about the Italian Prime Minister Renzi, collected on Twitter from April 20th to May 22nd 2015, with a hand-coded training set of $n = 1324$ texts. The texts were tagged according to the discussions about the Prime Minister's political action $D^{(1)}$ (from "Environment" to "School", $M^{(1)} = 10$ including Off-Topic) and according to the sentiment $D^{(2)}$ (Negative, Neutral, Positive and Off-Topic, $M^{(2)} = 4$) as shown in Table 14. The new variable D consists of $M = 25$ distinct and non-empty categories. Table 16 shows an example of the estimation of the joint distribution taking a sample of 182 texts to be used as training set. All texts such that the frequency of hand-coded tag is less than 10 in Table 14 are included in the sample; for those in which this number is greater than 10, we took a sample of 10 observations. Table 16 shows that iSAX and ReadMe perform almost the same, with a slightly higher preference for iSAX. RF, despite having the lowest MAE, has conversely a higher χ^2 value.

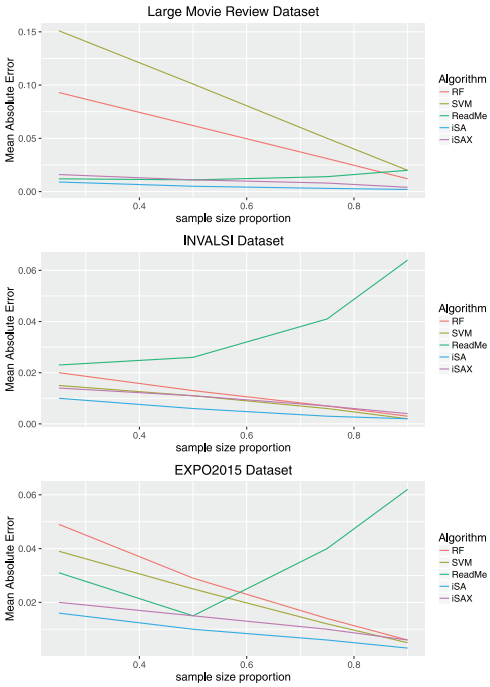

Fig. 3. Estimation results. Mean Absolute Error (left) and χ^2 statistic (right), averaged over the Monte Carlo replications, of the estimates of $P(D)$ for the different methods as a function of the sample size.

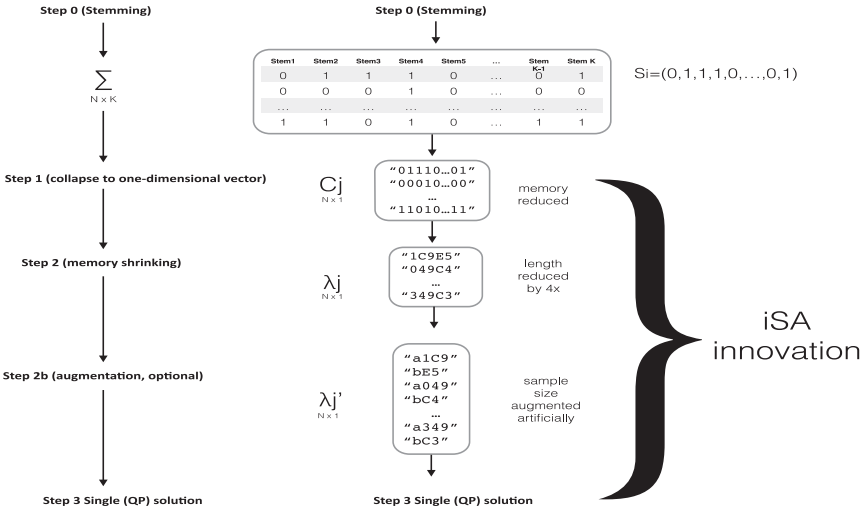

Fig. 4. The iSA workflow and innovation.

Table 15 shows the performance of iSAX on the whole corpus, based on a training set of 1324 hand-coded texts. The middle and bottom panel also show the conditional distributions, which are very useful in the interpretation of the analysis: for instance, thanks to the cross-tabulation we can observe that when people talk about the environmental issues or about law and order, Renzi attracts a relatively higher share of positive sentiment. Conversely, positive sentiment toward the Prime Minister is lower in conversations related to the state of the economy, labor policy and school reform. The results for the other methods are given in Table 17.

9. Discussion

After introducing the theoretical advantages of and reasoning for directly estimating the aggregated distribution of opinions rather than applying individual classification and subsequent aggregation of the estimates, we have presented a novel, fast and scalable algorithm called iSA (or iSAX) to perform the first task (aggregate estimation) which also produces sta-

ble (in term of variability) and accurate (in terms of Mean Absolute Error and χ^2-test) estimates of the target distribution of opinion $P(D)$. The iSA/SAX algorithm and its difference from the other approaches have been described in full technical detail.

We run Monte Carlo experiments using freely available and benchmark data sets as well as self collected data from a social network. The experimental results show that the iSA/iSAX algorithm is faster in almost all cases for large sample sizes, with respect to the competitors considered by a factor of 10x and up to 140x. It is also computationally efficient for small sample sizes. The iSA/iSAX algorithm has a constant execution time, where the time depends not on the number of stems used or the size of the training sample but on the whole size of the population of texts, i.e., training and test sets. Fig. 2 shows the computational time in seconds (left column) for each iteration of the Monte Carlo experiment and the relative increase in time spent for the computation (right column) as a function of the training sample size.

Accuracy and variability of the iSA/iSAX estimates decrease with sample size, as expected by standard asymptotic theory while, in some situations, other methods for aggregated estimation may increase both variability and MAE depending on the distribution of $P(D)$ in the training sample. Fig. 3 summarizes the results concerning the MAE (left) and shows a similar picture in terms of the χ^2 statistics (right).

Moreover, the standard error of the iSA/iSAX estimates can be easily obtained using a plain bootstrap analysis (even if not shown in all the tables). As for the ReadMe algorithm, the iSA algorithm is also scalable with respect to the number of stems considered for the analysis, while the other individual classifiers are affected in terms of computational times.

In addition to the above results, in Section 8 we have shown the way the can be cross-tabulated using all classifiers by a simple collapsing of the two-way table into a one dimensional vector of frequencies, for which any method should work if properly trained. In our example on the Renzi data set, we have shown a slight preference for iSAX and ReadMe over the other techniques.

Acknowledgments

The third author thanks the Graduate School of Mathematical Sciences, University of Tokyo, for his visiting professorship during which this study has been completed and CREST Program.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.ins.2016.05.052

References

[1] N. Aletras, J.H. Lau, T. Baldwin, M. Stevenson, Tm 2015 – topic models: post-processing and applications workshop, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, in: CIKM '15, ACM, New York, NY, USA, 2015, pp. 1953–1954, doi:10.1145/2806416.2806875.
[2] M. Bouchet-Valat, SnowballC: Snowball stemmers based on the C libstemmer UTF-8 library, 2014. R package version 0.5.1, URL: http://CRAN.R-project.org/package=SnowballC.
[3] L. Breiman, Random forests, Machine Learning 45 (1) (2001) 5–32.
[4] E. Cambria, B. Schuller, Y. Xia, C. Havasi, New avenues in opinion mining and sentiment analysis, IEEE Intell. Syst. 28 (2) (2013) 15–21.
[5] A. Ceron, L. Curini, S. Iacus, Social Media e Sentiment Analysis. L'evoluzione dei fenomeni sociali attraverso la Rete, Springer, Milan, 2013.
[6] A. Ceron, L. Curini, S. Iacus, Using sentiment analysis to monitor electoral campaigns. method matters. evidence from the United States and Italy, Soc. Sci. Comput. Rev. 33 (1) (2015) 3–20, doi:10.1177/0894439314521983.
[7] A. Ceron, L. Curini, S. Iacus, G. Porro, Every tweet counts? How sentiment analysis of social media can improve our knowledge of citizens political preferences with an application to Italy and France, New Media Soc. 16 (2) (2014) 340–358.
[8] L. Curini, S. Iacus, L. Canova, Measuring idiosyncratic happiness through the analysis of twitter: An application to the italian case, Soc. Indicators Res. 121 (2) (2015) 525–542.
[9] M.C. Frank, N.D. Goodman, Inferring word meanings by assuming that speakers are informative, Cogn. Psychol. 75 (2014) 80–96, doi:10.1016/j.cogpsych.2014.08.002.
[10] D. Hopkins, G. King, A method of automated nonparametric content analysis for social science, American J. Pol. Sci. 54 (1) (2010) 229–247.
[11] D. Hopkins, G. King, ReadMe: ReadMe: Software for Automated Content Analysis, 2013. R package version 0.99836, URL: http://gking.harvard.edu/readme.
[12] H.M. Iacus, Big data or big fail? the good, the bad and the ugly and the missing role of statistics, Electronic J. Appl. Stat. Anal. 5 (11) (2014) 4–11.
[13] S.M. Iacus, G. King, G. Porro, Multivariate matching methods that are monotonic imbalance bounding, J. American Stat. Assoc. 106 (2011) 345–361.
[14] E. Kalampokis, E. Tambouris, K. Tarabanis, Understanding the predictive power of social media, Internet Res. 23 (5) (2013) 544–559.
[15] G. King, Restructuring the social sciences: reflections from harvard's institute for quantitative social science, Polit. Pol. Sci. 47 (1) (2014) 165–172.
[16] A. Liaw, M. Wiener, Classification and regression by randomforest, R News 2 (3) (2002) 18–22. http://CRAN.R-project.org/doc/Rnews/
[17] Z. Lu, X. Wu, J.C. Bongard, Active learning through adaptive heterogeneous ensembling, IEEE Trans. Knowl. Data Eng. 27 (2) (2015) 368–381, doi:10.1109/TKDE.2014.2304474.
[18] A.L. Maas, R.E. Daly, P.T. Pham, D. Huang, A.Y. Ng, C. Potts, Learning word vectors for sentiment analysis, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Portland, Oregon, USA, 2011, pp. 142–150. http://www.aclweb.org/anthology/P11-1015
[19] D. Meyer, E. Dimitriadou, K. Hornik, A. Weingessel, F. Leisch, e1071: Misc Functions of the Department of Statistics (e1071), TU Wien, 2014. R package version 1.6-3, URL: http://CRAN.R-project.org/package=e1071.
[20] I. Sag, T. Baldwin, F. Bond, A. Copestake, D. Flickinger, Multiword expressions: a pain in the neck for nlp, in: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing, Lecture Notes in Computer Science, 2276, Springer Berlin Heidelberg, 2002, pp. 1–15, doi:10.1007/3-540-45715-1_1.
[21] H. Schoen, D. Gayo-Avello, P. Metaxas, E. Mustafaraj, M. Strohmaier, P. Gloor, The power of prediction with social media, Internet Res. 23 (5) (2013) 528–543.
[22] Voices from the Blogs, Expo2015: c'è ottimismo nel Mondo (scandali permettendo), 2014. June, 27, URL http://voicesfromtheblogs.com/2014/06/27/expo2015-ce-ottimismo-nel-mondo-scandali-permettendo/.

[23] Voices from the Blogs, L'Invalsi supera la prova Invalsi: migliora il giudizio (almeno in Rete), si copia di meno, ma i docenti..., 2014, June, 19, URL: http://voicesfromtheblogs.com/2014/06/19/linvalsi-supera-la-prova-invalsi-migliora-il-giudizio-almeno-in-rete-si-copia-di.
[24] J. Zhang, X. Wu, V.S. Sheng, Active learning with imbalanced multiple noisy labeling, IEEE T. Cybern. 45 (5) (2015) 1081–1093, doi:10.1109/TCYB.2014.2344674.

Andrea Ceron is an assistant professor of Political Science at the Department of Social and Political Sciences, Università degli Studi di Milano, in Milan. His research focuses on intra-party politics, quantitative text analysis, social media and political trust. His recent publications include, among others, articles in the British Journal of Political Science, European Journal of Political Research, Journal of Computer-Mediated Communication, New Media & Society, International Journal of Press/Politics and Party Politics.

Luigi Curini is an associate professor of Political Science at the Department of Social and Political Sciences, Università degli Studi di Milano and is a visiting associate professor at Waseda University, Tokyo. His main research interests focus on the spatial theory of voting, legislative behavior, quantitative methods, and social media analysis. His articles have appeared in the European Journal of Political Research, British Journal of Political Science, Comparative Political Studies, Journal of Politics, Public Choice, New Media & Society, and Social Science Computer Review. He is also author of several books.

Stefano M. Iacus is a full professor of Statistics at the Department of Economics, Management and Quantitative Methods, Università degli Studi di Milan, in Milan, Italy. He has been member of the R Core Team for the development of the R statistical environment (1999–2014) and author of many statistical packages for the R software. He is an associate editor of the J. of Stat. Software and J. Stat. Planning and Inference. He has been a visiting professor at Harvard University (USA), Le Mans University (France) and Tokyo University (Japan). He has published in mainstream journals of theoretical and computational statistics and is the author of two monographs.