Comput Manag Sci (2012) 9:401–415
DOI 10.1007/s10287-012-0149-z

ORIGINAL PAPER

Simple and efficient classification scheme based on specific vocabulary

Jacques Savoy · Olena Zubaryeva

Received: 27 June 2011 / Accepted: 18 June 2012 / Published online: 5 July 2012

Abstract Assuming a binomial distribution for word occurrence, we propose computing a standardized Z score to define the specific vocabulary of a subset compared to that of the entire corpus. This approach is applied to weight terms (character *n*-gram, word, stem, lemma or sequence of them) which characterize a document. We then show how these Z score values can be used to derive a simple and efficient categorization scheme. To evaluate this proposition and demonstrate its effectiveness, we develop two experiments. First, the system must categorize speeches given by B. Obama as being either electoral or presidential speech. In a second experiment, sentences are extracted from these speeches and then categorized under the headings electoral or presidential. Based on these evaluations, the proposed classification scheme tends to perform better than a support vector machine model for both experiments, on the one hand, and on the other, shows a better performance level than a Naïve Bayes classifier on the first test and a slightly lower performance on the second (10-fold cross validation).

Keywords Statistics in lexical analysis · Corpus linguistics · Text categorization · Machine learning · Natural language processing (NLP)

1 Introduction

During the last decade, various text categorization models and applications have been proposed (Weiss et al. 2010). As a first example related to this study, we find the

J. Savoy (✉) · O. Zubaryeva
Computer Science Department, University of Neuchatel,
Rue Emile Argand 11, 2000 Neuchâtel, Switzerland
e-mail: Jacques.savoy@unine.ch

O. Zubaryeva
e-mail: Olena.Zubaryeva@unine.ch

authorship attribution problem (Juola 2006) where, from a given sample of texts written by known authors, the author of a disputed text must be determined. Besides this particular problem other pertinent and related issues such as determining to the extent possible, any demographic or psychological information on the author (*profiling*) (Argamon et al. 2009). As a third example, political texts can be classified as either electoral or governmental speeches (Labbé and Monière 2003, 2010) or perhaps attribute them to a given party or ideology (Savoy 2010; Hirst et al. 2010). As a forth example related to opinion detection, text categorization approaches were applied to opinion-related information, and new text segments must be categorized as either opinionated or factual (Abassi et al. 2008; Boiy and Moens 2009). Moreover, Pang and Lee (2008) propose to identify the polarity of a written opinion (positive, negative or mixed). Instead of being limited to detection of opinion, we can identify sentiment and other private states (speculations, dreams, etc.).

In all of these text categorization examples, the corresponding text (e.g., sentence, paragraph, speech) is represented by a numerical vector comprising relevant features for distinguishing between various categories. Throughout this process we select those features that are useful for identifying differences in style between authors [authorship attribution (Stamatatos 2009)], between topics or categories (e.g., politics, finance, macro-economics, sport) (Sebastiani 2002) or between genres (survey, editorial, research paper, blogs, homepage, etc.) (Kanaris and Stamatatos 2009). In a second stage, we weight them according to their discriminative power. Finally, through using a set of classification rules or a learning scheme, the system has to decide whether or not to assign a single category to each input text (*single-label categorization* problem).

To achieve this objective, we propose and evaluate a new and simple method for weighting the terms used to represent documents within a text categorization system. Our approach is mainly based on differences found between an expected occurrence frequency and the observed occurrence frequency of terms within two disjoint subsets. Based on a standardized Z score, we define over-used terms in one subset (defined as its specific vocabulary), terms common to both subsets (the common vocabulary) and finally under-used terms. Finally, in our opinion a simple categorization rule producing a reasonable performance level is better than a complex categorization strategy viewed as a black box by the final user.

The rest of this paper is divided as follows. Section 2 presents related work, while Sect. 3 depicts the main characteristics of the corpus used in our experiments. Section 4 briefly describes certain text categorization approaches upon which our experiments were based. This section also presents our suggested categorization model based on the Z score method. Section 5 evaluates these models by applying them to two different text categorization problems, and our main findings are summarized in the last section.

2 Related work

During the last decade important advances have been made in the field of text categorization (Sebastiani 2002). In this kind of applications the underlying idea is to assume that there were distinct structural or statistical relationships between features (such as

sequence of n characters (or n-grams), words, word sequences, lemmas, isolated or sequence of part-of-speech (POS) tags, phrases) and the corresponding categories.

Among all possible features, the most important objective is to select those that can reflect the style of the corresponding text-category, genre or author. First, at the lexical level, we have considered word occurrence frequency, average word length, letter occurrence frequency (Merriam 1998), frequency of punctuation, along with other symbols, etc. In authorship attribution in particular, special attention has been paid to the use of very frequent words (e.g., *a*, *the*, *in*, *of*, *and*, *you*, *is*, *must*). This choice is motivated by the assumption that a given author's style is better reflected by the use of such very frequent forms than the use of more topic-oriented terms. Although the precise definition of these common word lists is questionable, authors have suggested a wide variety of lists. Burrows (2002) suggests a list of 150 terms while Hoover (2006) put forward a list of more than 1,000 frequently occurring words.

Secondly, at the syntactic level, we could account for POS information through measuring the distribution, frequency, patterns or combinations of these items (Stamatatos 2009).

Thirdly, various authors (Zheng et al. 2006; Kanaris and Stamatatos 2009; Stamatatos 2009) have also suggested considering structural and layout features including the total number of lines, number of lines per sentence or per paragraph, paragraph indentation, presence of greetings or particular signature formats, as well as features derived from HTML tags.

In order to evaluate the relative merits of these possible feature sets, Finn and Kushmerick (2006) have used a decision-tree approach (C4.5 algorithm) to classify documents according to their topics (football, politics, and finance). They investigated three feature-sets. The first representation was a bag-of-words approach based on the presence or absence of words, after stopword removal and stemming (Porter 1980). In a second experiment, each document was represented by a vector of 36 POS tags (e.g., determiner, adverb, verb base form, modal, foreign word, etc.). This second model tried to reflect the intrinsic stylistic patterns while the first, depending on word usage, was better to reflect the underlying semantics. In a third experiment, Finn and Kushmerick (2006) suggested computing three document-level statistics (number of words, sentence length, word length), along with the frequency of 117 very frequent words (e.g., *because*, *did*, *each*, *large*, *us*, etc.), and 17 punctuation symbols. The accuracy resulting from these three representations indicates that the bag-of-words tended to perform better, even though the performance differences with the other two representations were small.

The text genre might impose a given style and new communication media favour certain forms of writing and layout, such as the overuse of uppercase letters frequently found in spam e-mails. As another example, we could discriminate between various webpage genres (e.g., personal homepage, corporate homepage, blogs, e-shops, etc.). In this case, Kanaris and Stamatatos (2009) suggested accounting for HTML tags and variable length character n-grams (with $3 \leq n \leq 5$). Moreover, various internet-based writing situations (e.g., e-mail, chat groups, instant messaging, blogs, virtual worlds, etc.) generate their own literary register due to their specific graphical, orthographical, lexical, grammatical and structural features (Crystal 2006). Examples of these include the presence of smiley's (e.g., :-)), the use of commonly occurring abbreviations

Table 1 Statistics on our US political corpus

		Electoral	Presidential
1	Number of speeches	113	167
2	Number of tokens	340,068	343,388
3	Number of distinct word types	8,698	12,374
4	Number of distinct lemmas	7,193	9,720
5	Mean number of lemmas/speech	2,891.4	2,021.5
6	Median number of lemmas/speech	2,830	1,605
7	Mean number of distinct lemmas/speech	695.8	552.8
8	Number of sentences	10,724	11,140
9	Mean number of lemmas/sentence	27.5	27.4
10	Median number of lemmas/sentence	24	24
11	Mean number of distinct lemmas/sentence	23.0	22.7

(e.g., *irl* for *in real life*), as well as certain graphical conventions (e.g., emphasis created with uppercase letters, spaces or stars such as *I SAID N O !*), together with the possible presence of various colors and animations in the displayed documents.

3 Categorization tasks

Our main objective is to investigate the style differences between electoral and presidential speeches. To achieve this, we have selected the political speeches given by B. Obama during the last year of his presidential campaign (2008) and during the first 6 months of his administration (2009). We chose this political thematic for various reasons. First, these documents are easy to obtain and of adequate quality (correct spelling, consistent and simple encoding). Second, they are written in the same language (American English in our case) within a short period of time, and they refer to the same background material. Third, the choice of vocabulary and syntax is not arbitrary but rather motivated by the underlying objectives of each speech.

The evaluation corpus is divided into a first group of 113 electoral speeches from the year 2008, while in the second we included the first 167 speeches given by the new President, from 20 January (investiture speech) to 23 July in 2009. All these texts were downloaded from their official web sites (http://www.BarackObama.com and http://www.WhiteHouse.gov).

For each speech, we replace certain system punctuation marks in UTF-8 coding with their corresponding ASCII code symbols, and removed a few diacritics found in certain words (e.g., communiqué or Chávez).

As depicted in the first two lines of Table 1, even though the number of speeches is distinct (113 vs. 167), the number of word tokens (with punctuation and numbers) is similar (340,068 vs. 343,388). From these values we can infer that on average, electoral speeches were longer than the presidential addresses (as shown in Table 1 by the precise values listed in the middle). In this table, we can also find the number of

word types (or distinct words) and the number of lemmas (headwords or dictionary entries). When counting the number of word types, all inflected forms have their own entry (e.g., *is*, *was*, *be* or *soldiers*, *soldier*), and when counting lemmas, all forms are grouped under the same dictionary entry (e.g., *be* or *soldier* in our previous example). Clearly the number of features is reduced when using lemmas instead of word tokens.

We then plan two experiments based on this corpus. In the first we must classify the speeches as either *electoral* or *presidential*. As shown in rows five through seven, the mean number of lemmas per speech ranged from 2,891.4 (median 2,830) during the elections to 2,021.5 (median 1,605), for a typical presidential speech.

In a second experiment, we must classify the sentences extracted from these speeches as either *electoral* or *presidential*. Instead of considering all sentences, we removed any sentences having fewer than ten elements (e.g., sentences composed of only one word such as in the sequence "Yes. We can.", or very short sentences such as "Good afternoon". were ignored). As depicted in the bottom part of Table 1, the mean number of lemmas is relatively small (27.5) compared to the mean speech length. As such the mean number of possible features used to discriminate between *electoral* or *presidential* sentences is significantly reduced and thus when replacing speeches by sentences we should expect a decrease in classification quality.

4 Text classification models

To design and implement an automatic text classification system we needed to represent the texts to be classified and to specify the classifier model. The following section describes the representation used in our experiments. We compare three classifiers: the Naïve Bayes (Sect. 4.2), the support vector machine (SVM) (Sect. 4.3), and a new approach based on the Z score (Sect. 4.4).

4.1 Text representation and feature selection

Text representations are usually based on words. Although a word's definition is always difficult to establish along with the necessary generality, we define a word as a sequence of letters and digits beginning with a letter (e.g., *PDP11* should be viewed as one word while the string *PDP-11* will be reduced to *PDP*). Moreover, we know that the surface forms may vary according to syntactical (e.g., *do*, *does*, *doing*) and morphological rules (e.g., *biological* vs. *biology*) even though they represent to a very similar meaning. As such, we represent text based on the inflected forms (e.g., *armies*), the stems (*army* or even *arm*), or the lemmas (*army*/*noun*). A recent study (Fautsch and Savoy 2009) demonstrated that no significant performance differences were found between representations based on a light stemmer, an aggressive one, or lemmas, at least in the information retrieval domain and with the English language.

Based on these considerations we decided to base our textual representation on lemmas, and thus to group all inflected forms under the same entry (e.g., *chose*, *choose*, *chosen* into *choose*). To automatically assign the corresponding POS tag to each word we used the POS tagger developed by Toutanova et al. (2003). As a result, each word type received a POS tag and some morphological information. For example, from

the sentence "After hearing about the good reviews, Clinton reportedly showed interest in it." the POS tagger will return "After/IN hearing/VBG about/IN the/DT good/JJ reviews/NNS,/, Clinton/NNP reportedly/RB showed/VBD interest/NN in/IN it/PRP ./.". In this sequence, we might find tags attached to nouns (NN, noun, singular, NNS noun, plural, NNP proper noun, singular), to verbs (VB, base form or lemma, VBP non-3rd-person singular present, VBZ 3rd-person singular present, VBN past participle), adjectives (JJ, JJR adjective in comparative form), personal pronouns (PRP), possessive pronouns PRP, prepositions (IN) and adverbs (RB). Using this information we can then derive the lemma by removing, for example, the plural form for nouns (e.g., keys/NNS → key/NN).

As with all text categorization problems, we are facing with a high dimensional feature space, and not all features (or lemmas) are very useful for discriminating between the distinct categories. As a first dimension reduction scheme, we do not represent text by all the inflectional words but by lemmas. As shown in Table 1 and for the presidential speeches, the number of features was reduced from 12,374 possible word types to 9,720 lemmas, a reduction of around 21.45 %.

As a second feature's selection procedure, we will use the document frequency statistics, considered as a useful relevance indicator in information retrieval (Manning et al. 2008). Using document frequency as selection feature was also found effective in other text categorization problems (Yang and Pedersen 1997). In the current study, we have removed all lemmas having a document frequency of three or less. Upon inspecting the presidential speeches this pruning scheme reduces the dimensional feature space to 3,746 (a reduction of 59.6 % compared to the starting size of 12,374).

4.2 Naïve Bayes

As a first baseline we adopt the classical Naïve Bayes model (Mitchell 1997), where the system must select for each text between two hypotheses, namely h_0 (presidential) and h_1 (electoral). In our experiments, the set of possible texts is either the speeches or the sentences. The selected category corresponds to the one maximizing Eq. (1), where *m* indicates the number of lemmas (over all possible lemmas) included in the current text (either a speech or a sentence), and t_j the different terms (lemmas) belonging to this speech or sentence representation.

$$\underset{h_i}{\text{Arg max}} \; \text{Prob}[h_i] \cdot \prod_{j=1}^{m} \text{Prob}[t_j \mid h_i] \tag{1}$$

The underlying probabilities still had to be estimated. For the prior probabilities $\text{Prob}[h_i]$, the estimate is simply based on the proportion of each category (e.g., when classifying speeches, $\text{Prob}[h_{\text{electoral}}] = 113/(113 + 167) = 0.4036$). To estimate the probabilities related to the different terms, we group all speeches (or sentences) belonging to the same category (set denoted T_{hi}). The corresponding probabilities are then estimated according to Eq. (2). This formulation corresponds to the ratio between the number of occurrences of term t_j in the set T_{hi} (denoted n_{ji}) and the number of all occurrences in the whole T_{hi} set (e.g., all electoral speeches) denoted n_i.

Table 2 The ten most frequent lemmas along with their number of occurrences and estimated probabilities in the presidential and electoral speeches

	President			Candidate		
	Prob. ($\times 10^5$)	n_{0j}	Lemma	Prob. ($\times 10^5$)	n_{1j}	Lemma
1	425.3	13,093	the	416.8	12,429	the
2	381.0	11,731	be	355.2	10,591	and
3	357.1	10,994	and	341.6	10,185	be
4	331.3	10,200	to	319.4	9,524	we
5	319.8	9,847	we	292.5	8,721	to
6	255.9	7,879	of	228.4	6,810	that
7	234.8	7,230	that	226.6	6,757	of
8	208.3	6,413	an	201.5	6,008	an
9	168.8	5,198	in	171.3	5,108	in
10	140.8	4,336	I	167.9	5,005	I

$$\text{Prob}[t_j \mid h_i] = n_{ji}/n_i \tag{2}$$

Based on the maximum likelihood principle, this calculation tends to over-estimate the probabilities of terms occurring in the text at the expense of the missing terms. For the latter, the occurrence frequency is 0 and the corresponding probability is also 0. It is known however that the word distribution behaves according to the LNRE law [*Large Number of Rare Events* (Baayen 2001)], so as a correction we suggest applying a simple smoothing technique and thus eliminating the special processing problem when an occurrence probability is 0.

As a first approach, Laplace suggests adding one to the numerator in Eq. (2) and likewise adding the vocabulary size to the denominator (Manning and Schütze 2000). This approach can be generalized (Lidstone's law) through smoothing each probability estimate as $\text{Prob}[t_j \mid h_i] = (n_{ji}+\lambda)/(n_i+\lambda \cdot |V|)$, with λ a parameter, and $|V|$ indicating the vocabulary size (see Table 1 for their values). In our experiments, we set this value to 0.1 because we believe a larger value would assign unnecessarily large probabilities to rare words. Finally, when compared to the Good–Turing approach (Sampson 2001), this smoothing technique is rather easy to implement.

When using our US political corpus, Table 2 shows the ten lemmas having the highest estimated probabilities in the presidential or electoral subsets. These estimations are based on the number of occurrences (under the labels n_{0j} and n_{1j}) and a smoothing factor $\lambda = 0.1$. As shown, the same terms occur in both parts, and in more or less the same order. Based on a comparison of these two lists, the main differences appear in the probability estimates and not on the presence or absence of the terms.

4.3 Support-vector machine (SVM)

As a second approach we used the SVM model (Joachims 2002), which usually performed well on various text categorization tasks. In this model, a term vector represents

each text, and to reflect its importance in the underlying representation a weight is assigned to each term. Derived from the information retrieval vector-space model, a classical technique is to weight each term through applying the *tf idf* formula (Manning et al. 2008), in which the component *tf* represents the number of occurrences within the text. The $idf (=\log(n/df))$ corresponds to the logarithm of the inverse document frequency (denoted *df*), and thus indicates the number of texts in which the corresponding term occurs, while n indicates the total number of texts in the corpus.

As an alternative, we might normalize both components so that the only possible values would fall between 0 and 1. For the *tf* part, we select the augment *tf* (or *atf*) weighting scheme defined as $atf = 0.5 + 0.5 \cdot (tf/max\ tf)$, where the value *max tf* corresponds to the maximal number of occurrences for the corresponding text and *idf* is obtained by simply dividing the *idf* value by $\log(n)$ (normalization denoted *nidf*).

Based on this representation we used the freely available SVM^{light} model (Joachims 2002) which determines the hyperplane that best separates the examples belonging to the two categories. In this case *best* hyperplane refers to that having the largest separation (or margin) between the two classes (together with the reduction of the number of incorrect classifications). This first version belongs to the linear classifier paradigm and we can also consider non-linear kernel functions (polynomial, sigmoid). When based on a *tf idf* or an *atf nidf* text representation, these advanced discrimination functions did not improve the quality of the classification effectiveness, at least in our classification tasks.

4.4 Z-score based classification model

As a new classification approach, we suggest basing the weighting of terms (words, lemmas) using the specific vocabulary approach proposed by Muller (1992). To define the term's specificity, we split the entire corpus into two disjoint sets denoted h_0 and h_1 (e.g., electoral vs. presidential speeches). For a given term t_j we count the number of occurrences in the set h_0 (value denoted n_{j0}) and the number of occurrences in the second part h_1 (denoted n_{j1}). Thus, for the entire corpus, we have $n_{j0} + n_{j1}$ occurrences of the corresponding term. The total number of all occurrences in the set h_0 is denoted by n_0, similarly for n_1 indicating the size of part h_1, and $n = n_0 + n_1$ corresponds to the size of the whole corpus.

The Z score model is based on the following idea. When selecting randomly a lemma from a corpus, we can draw the term t_j or another (Bernoulli process). We can repeat, with replacement, this drawing n_0 times and ask the number of occurrences of the target term t_j we can obtain. In such case, the distribution of the term t_j follows a Binomial distribution with parameters n_0 and $\text{Prob}[t_j]$ representing the probability of randomly selecting the term t_j from the entire corpus. Based on the maximum likelihood principle, this probability would be estimated as $(n_{j0} + n_{j1})/n$. As described previously, to obtain a better estimation we apply the Lidstone's smoothing rule. To estimate the expected number of occurrences of the term t_j in the set h_0, we use the expression $n_0 \cdot \text{Prob}[t_j]$. We could then compare this expected number to the observed number (namely n_{j0}), and any large difference between these two values would indicates a deviation from the expected behavior. To obtain a more precise definition

Table 3 The ten lemmas with the highest and smallest Z score along with their number of occurrences in our political corpus

	President			Candidate		
	Z score	n_{0j}	Lemma	Z score	n_{1j}	Lemma
1	12.6	646	thank	−18.3	710	McCain
2	10.6	243	everybody	−15.2	872	tax
3	10.3	239	recovery	−13.0	486	Street
4	9.8	478	reform	−12.5	528	Senator
5	8.0	193	extraordinary	−12.2	331	Bush
6	7.8	2,251	so	−11.1	366	election
7	7.4	263	folk	−10.6	357	Wall
8	7.3	1,278	these	−10.4	6,580	not
9	7.2	436	very	−9.8	737	Washington
10	6.6	19,765	to	−9.8	937	change

of large we could account for the variance within the underlying Binomial process [defined as $n_0 \cdot \text{Prob}[t_j] \cdot (1 - \text{Prob}[t_j])$]. Then we compute the standardized Z score defined in Eq. (3) for each lemma t_j and according to the presidential subset h_0.

$$\text{Z score}(t_j, h_0) = \frac{n_{j0} - n_0 \cdot \text{Prob}[t_j]}{\sqrt{n_0 \cdot \text{Prob}[t_j] \cdot (1\text{-Prob}[t_j])}} \tag{3}$$

Based on the Z score value for a given term, we can verify whether it is used proportionally with roughly the same frequency in both parts (Z score value close to 0). On the other hand, when a term has a positive Z score larger than a given threshold δ (e.g., $\delta = 2$), we consider it as being over-used in the presidential subset or belonging to the specific vocabulary of part h_0. A large negative Z score (less than $-\delta$) indicates that the corresponding term is under-used in the set h_0 (or similarly over-used in the electoral subset h_1).

Based on the information depicted in Table 3, we can analyze the most important lexical differences between the electoral and presidential speeches. In the specific vocabulary belonging to the presidential discourse could be found *recovery* (score Z = 10.3), *reform* (9.8), *extraordinary* (8.0), *folk* (7.4), or *very* (7.2) while for terms specific to electoral speeches (and under-used in presidential speeches), we could found *McCain* (score Z = −18.3), *tax* (−15.2), *Street* (−13.0), *Bush* (−12.2) or *change* (−9.8). During the election campaign, B. Obama often uses the terms *Senator McCain* or *Senator John McCain* to refer to his opponent, and employs the words *Wall Street* and *Main Street* to indicate the finance or the real economic world (e.g., "there is no dividing line between Wall Street and Main Street"). Moreover, as a candidate he also need to repeat his arguments to convince the electorates (Bush administration, Bush tax cut, time for change), while as President he demonstrates his will to reform the economics through introducing specific vocabulary changes (new foundation, reform,

Table 4 Example of a contingency table for evaluating a binary classifier

Decision taken	Correct decision	
	Category A	Category $\neg A$
Category A	True positive	False positive
Category $\neg A$	False negative	True negative

recovery). He also needs to refer indirectly to an undetermined set of persons (e.g., with the lemma *folk* in the sentence "the same folk who are making these criticisms,").

To define a classification rule based on the Z score, our first attempt involved accounting for the number of Z scores that were higher or lower than the given threshold δ. For a given speech or sentence, we could count the number of large and positive Z scores (over-used terms), and similarly the number of large negative Z scores (under-used terms). As a decision rule, we could assign the category corresponding to the largest number of terms either as over-used (category h_0) or under-used (category h_1). This rule will however attach the same importance to a term having a Z score value just above the threshold (e.g., 1.1 with $\delta = 1$) compared to a very specific term having a higher Z score value (e.g., 3.5).

When using this first strategy we still need to specify a parameter (the value for the threshold δ). Moreover, this rule does not account for the prior distribution, or the fact that one category might be more frequent than the other. Finally when facing with a short text excerpt to be classified, the final decision can be taken based on a very limited number of terms (and even a single one) over-used or under-used. To overcome these problems we simply suggest to sum all positive Z scores on the one hand, and on the other all negative Z scores. As a decision-making rule we simply specify the category corresponding to the largest sum, and used the prior distribution to break ties.

5 Evaluation

5.1 Evaluation measures

In various text categorization studies, performance evaluation is based on precision, recall and F-measures, all three measures related to the two error types a system could produce. To illustrate their definitions, Table 4 illustrates the four possible outcomes of binary decisions (e.g., Category A or ¬A). The *true positive* and *true negative* cells show the number of correct decisions made by the system. The *false positive* cell indicates the number of texts classified by the system under "Category A" when the correct decision is the opposite. Finally the *false negative* cell contains the number of texts belonging to "Category A" but classified by the system as "Category ¬A".

$$\text{Precision } \pi = \frac{\#\text{ true positive}}{\#\text{ true positive} + \#\text{ false positive}} \tag{4}$$

$$\text{Recall } \rho = \frac{\#\text{ true positive}}{\#\text{ true positive} + \#\text{ false negative}} \tag{5}$$

Based on this contingency table, Eq. (4) defines precision as the percentage of texts the system has correctly classified under Category A. Equation (5) defining the recall represents the percentage of "Category A" texts correctly classified under this class. When comparing two text categorization models, both measures could possibly generate problematic situations (e.g., one system could show a better precision while the other better recall). In order to obtain a single measure reflecting the system's ability to return correct results only for both precision and recall results, we use the $F_{(\beta=1)}$ measure described in Eq. (6). In the current study we have fixed $\beta = 1$, and then we attach the same importance to both precision and recall (harmonic mean).

$$F_{(\beta)} = \frac{(1+\beta^2)\cdot\pi\cdot\rho}{\beta^2\cdot\pi+\rho} \quad \text{with} \quad F_{(1)} = \frac{2\cdot\pi\cdot\rho}{\pi+\rho} \tag{6}$$

Finally to ensure unbiased precision, recall and $F_{(1)}$ measures, the same texts used to estimate the classification scheme's probabilities or parameters cannot be used in the evaluation stage. To measure the system's effectiveness we thus adopted a tenfold cross-validation method in which the kth fold is reserved for testing and the rest of the $k-1$ folds for training (Hastie et al. 2009). From the k measures obtained, we simply compute the arithmetic mean to define either precision, recall or $F_{(1)}$. It is important to note that in all our runs we have used the same subdivisions in k folds, with the same texts being placed in the different folds. Based on this fixed partition and to determine whether there are any statistical significant differences between two means, we can apply the paired t test with a significance level set at 5 % (Grimm 1993).

5.2 Evaluation of political speeches

In our first experiment we consider the electoral and presidential speeches given by B. Obama. It is our opinion that both types of speeches have their own specific characteristics, and thus for a particular speech the computer should be able to detect whether its text could be classified under the electoral or presidential category. In a recent study Labbé and Monière (2010) showed that both speeches types possess certain distinct features. For example during the last Canadian electoral campaign (2010), the former Prime Minister tended to use more forms related to the country (e.g., *Canadian*, *Canada*), and more frequently used the pronoun *we* and negative sentences. By contrast, an inspection of his governmental speeches reveals that more nouns and more names and fewer verbal forms, and future forms (e.g., *will*) could be included. Upon comparing the latest US presidential campaign and speeches given by the two candidates, Savoy (2010) drew similar conclusions. Obama's electoral speeches tend also to include more negative expressions with the relatively high frequency of the lemma *not* (occurring in rank 13th in Table 2).

This first experiment is based on 113 electoral and 167 presidential speeches. Table 5 lists the resulting precision, recall and F-measures obtained from our three classifiers along with their various parameter settings. When setting our parameters we specified the smoothing value λ (e.g., $\lambda = 0.1$) and the minimal document frequency

Table 5 Evaluation of various classifiers strategies based on 113 electoral and 167 presidential speeches (10-fold cross-validation)

		Precision	Recall	$F_{(1)}$
1	Naïve Bayes, $\lambda = 0.1$, min 4	54.88 %†	63.72 %†	58.74 %†
2	Naïve Bayes, $\lambda = 1$, min 4	54.88 %†	63.72 %†	58.74 %†
3	Naïve Bayes, $\lambda = 0.1$, min 0	60.05 %†,*	79.70 %†,*	68.37 %†,*
4	Naïve Bayes, $\lambda = 0$, min 4	54.88 %†	63.72 %†	58.74 %†
5	SVM, *tf idf*, min 4	65.58 %†	100.0 %†	79.10 %†
6	SVM, *tf idf*, min 0	66.05 %†	100.0 %†	79.40 %†
7	SVM, *atf nidf*, min 4	64.46 %†	100.0 %†	78.31 %†
8	Z Score, $\lambda = 0.1$, min 4	84.82 %	99.38 %	91.34 %
9	Z Score, $\lambda = 1$, min 4	60.97 %*	100.0 %	75.73 %*
10	Z Score, $\lambda = 0.1$, min 0	84.82 %	99.38 %	91.34 %
11	Z Score, $\lambda = 0$, min 4	85.53 %	98.75 %	91.52 %

(e.g., min 4) in order to be taken into consideration during the classification task. Thus a minimum document frequency value of four ensured that all terms appearing in three or less documents will be ignored. For the SVM model we have also specified the weighting scheme to be used (either *tf idf* or *atf nidf*).

Based the overall F-measure, it become evident that the Z score scheme obtained the best performance value of 91.52 % (Model #11, $\lambda = 0$, and min 4). Ranking second was the SVM approach, with a F-measure of 79.4 % (Model #6, *tf idf*, min 0), and finally the Naïve Bayes model (performance of 68.37 %, Line #3, $\lambda = 0.1$, min 0).

To verify whether the performance differences resulting from the various models are statistically significant, we selected Model #8 as the baseline and applied a paired *t* test (significance level 5 %) against all the others. The symbol "†" will be added to identify all statistically significant differences. As shown in Table 5, the differences are always significant for the F-measure when comparing the Z score Model #8 and the other Naïve Bayes or SVM approaches. The same conclusion can be drawn for both precision and recall, while for the latter the SVM approaches tend to perform better than the baseline Model #8 (Z score, $\lambda = 0.1$, min 4).

Other interesting verifications involved the performance differences within the same model but for different parameter settings. This was done by selecting the first line of each model as baseline (namely, Lines #1, #5 and #8) and the "*" symbol to identifying all statistical differences with this baseline performance. The data depicted in Table 5 indicates two significant differences, namely between Model #1 and #3 (Naïve Bayes), and between Model #8 and #9 (Z score).

Based on this statistical analysis we are able to deduce that the different parameter settings tended to produce similar performance levels within the SVM models. For the Naïve Bayes model, reducing the features space (min 4) seems to reduce the overall performance levels (Model #1 vs. #3). When inspecting the Z score models, it seems that a relatively large value for the parameter λ (e.g., $\lambda = 1$) might hurt the overall performance (Model #8 vs. #9).

Table 6 Evaluation of various classifiers strategies, US corpus (11,140 presidential vs. 10,724 electoral sentences, 10-fold cross-validation)

		Precision	Recall	$F_{(1)}$
1	Naïve Bayes, $\lambda = 0.1$, min 4	78.33 %†	78.50 %†	78.22 %†
2	Naïve Bayes, $\lambda = 1$, min 4	78.97 %†,*	77.10 %*	77.82 %†,*
3	Naïve Bayes, $\lambda = 0.1$, min 0	77.68 %†	79.87 %†,*	78.56 %†
4	Naïve Bayes, $\lambda = 0$, min 4	78.31 %†	78.71 %†,*	78.31 %†,*
5	SVM, *tf idf*, min 4	51.02 %†	100.0 %†	67.35 %†
6	SVM, *tf idf*, min 0	50.50 %†,*	100.0 %†	66.90 %†,*
7	SVM, *atf nidf*, min 4	51.34 %†	100.0 %†	67.65 %†,*
8	Z Score, $\lambda = 0.1$, min 4	70.76 %	76.43 %	73.27 %
9	Z Score, $\lambda = 1$, min 4	62.06 %†,*	91.95 %†,*	74.03 %
10	Z Score, $\lambda = 0.1$, min 0	70.85 %*	76.47 %	73.35 %
11	Z Score, $\lambda = 0$, min 4	71.99 %*	74.08 %*	72.80 %*

5.3 Evaluation of political sentences

To define our second classification task, we extracted sentences from either the electoral or presidential speeches, and then ask the system to classify them as electoral or to presidential. In this process we removed sentences having less than ten words. Our corpus is composed of 10,724 electoral and 11,140 presidential sentences. The mean number of features per sentence is rather small (around 27, as shown in Table 1), compared to around 2,000 terms for the speeches. This categorization task can thus be viewed as more complex and we expect lower performance levels.

Upon applying the three classifiers with different parameter settings, we obtained the results depicted in Table 6. In this case, the Naïve Bayes model produces the best F-measure (e.g., Model #3 with a $F_{(1)}$ value of 78.56 %), while the Z score model tends to show slightly lower precision and F-measure levels. The SVM model's overall $F_{(1)}$ performance level is lower than the others. To verify whether these performance differences are statistically significant we select the Z score Model #8 ($\lambda = 0.1$, min 4) as baseline and compare its performance with all the others. All statistically significant differences are denoted by the symbol "†". This set of tests indicates that the performance differences between the Z score and the Naïve Bayes schemes were statistically significant, as are the differences between the Z score and SVM schemes.

Finally to verify whether different parameter settings would significantly modify performance levels, we select the first line of each model as baseline (namely, Model #1, #5 and #8) and then compare their performance with other models derived from the same family. In Table 6 all statistical performance differences are denoted by the symbol "*". In this table we can see that for the Naïve Bayes approach, there are mainly significant differences between Model #1 ($\lambda = 0.1$, min 4) and #2 ($\lambda = 1$, min 4), indicating that a relatively large value for the parameter λ is not the best choice. For the SVM paradigm, we mainly detect a significant difference between Model #5 (*tf idf*, min 4) and #6 (*tf idf*, min 0), showing that ignoring lemmas appearing in less

than four texts tends to improve slightly the performance. For the Z score significant performance differences were revealed between Model #8 ($\lambda = 0.1$, min 4) and #9 ($\lambda = 1$, min 4). A larger value for the parameter λ tends to increase the recall and hurt the precision, producing a null effect on overall $F_{(1)}$ performance. As a general trend, we do not see any real and important effect when varying the different parameter settings within these three classifier models.

6 Conclusion

In this paper we describe a new and simple classification scheme based on the Z score values of lemmas (Muller 1992). Based on this definition, we determine terms over-used by B. Obama as presidential candidate (*McCain*, *tax*, *Street*, *Bush*, or *change*) compared to those he over-used as President (*recovery*, *reform*, *extraordinary*, *team*, or *budget*). Based on Z score values, we propose a new classifier and demonstrate its effectiveness in categorizing political speeches as either electoral or presidential (see Table 5). The Z score approach produces an F-measure (91.34 %) clearly superior to those of both the SVM model (79.40 %) and the Naïve Bayes approach (68.37 %). For classifying sentences however the Naïve Bayes model (see Table 6) proves to be the best overall performance (F-measure 78.56 %), while the Z score results in a slightly lower performance level (74.03 %), and the SVM model clearly produces the lowest performance level (67.65 %), at least for this task. For the Z score-based and Naïve Bayes models, ignoring those terms whose document frequency are three or less does not drastically modify performance levels, except when classifying speeches with the Naïve Bayes model (see Table 5), in which case the recall and F-measure values increase when considering all terms.

Finally, when using lemmas to represent text excerpts we must verify whether character *n*-grams along with other isolated word-based representations (e.g., inflected word forms, stems) would provide similar performance levels. Our experiments were based on a political context using the English language. We need to verify whether similar performance levels could be achieved when applied to other natural languages and other contexts (e.g., classifying incoming emails as pertinent or spam or as having a high or low priority).

Acknowledgments The authors would like to thank the anonymous referees for their helpful suggestions and remarks. This research was supported in part by the Swiss NSF under Grant #200021-124389.

References

Abassi A, Chen H, Salem A (2008) Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. ACM Trans Inf Syst 26(3)

Argamon S, Koppel M, Pennebaker JW, Schler J (2009) Automatically profiling the author of an anonymous text. Commun ACM 52(2):119–123

Baayen HR (2001) Word frequency distributions. Kluwer Academic Press, Dordrecht

Boiy E, Moens M-F (2009) A machine learning approach to sentiment analysis in multilingual Web texts. Inf Retr 12(5):526–558

Burrows JF (2002) Delta: a measure of stylistic difference and a guide to likely authorship. Lit Linguist Comput 17(3):267–287

Crystal D (2006) Language and the Internet. The Cambridge University Press, Cambridge
Fautsch C, Savoy J (2009) Algorithmic stemmers or morphological analysis: an evaluation. J Am Soc Inf Sci Technol 60(8):1616–1624
Finn A, Kushmerick N (2006) Learning to classify documents according to genre. J Am Soc Inf Sci Technol 57(11):1506–1518
Grimm LG (1993) Statistical applications for the behavioural sciences. Wiley, New York
Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning. Data mining, inference, and prediction. Springer, New York
Hirst G, Riabinin Y, Graham J (2010) Party status as a confound in the automatic classification of political speech by ideology. In: Proceedings JADT-2010, Rome, pp 731–742
Hoover DL (2006) Stylometry, chronology and the styles of Henry James. Digital Humanities, pp 78–80
Joachims T (2002) Learning to classify text using support vector machines. Methods, theory and algorithms. Kluwer, London
Juola P (2006) Authorship attribution. Found Trends Inf Retr 1(3)
Kanaris I, Stamatatos E (2009) Learning to recognize webpages genres. Inf Process Manag 45(5):499–512
Labbé D, Monière D (2003) Le discours gouvernemental. Canada, Québec, France (1945–2000). Champion, Paris
Labbé D, Monière D (2010) Quelle est la spécificité des discours électoraux? Le cas de Stephen Harper. Can J Political Sci 43(1):69–86
Manning CD, Schütze H (2000) Foundations of statistical natural language processing. MIT Press, Cambridge
Manning CD, Raghavan P, Schütze H (2008) Introduction to information retrieval. Cambridge University Press, Cambridge
Merriam T (1998) Heterogeneous authorship in early Shakespeare and the problem of Henry V. Lit Linguist Comput 13:15–28
Mitchell TM (1997) Machine learning. McGraw-Hill, New York
Muller C (1992) Principes et méthodes de statistique lexicale. Honoré Champion, Paris
Pang B, Lee L (2008) Opinion mining and sentiment analysis. Found Trends Inf Retr 2(1–2)
Porter MF (1980) An algorithm for suffix stripping. Program 14(3):130–137
Sampson G (2001) Empirical linguistics. Continuum, London
Savoy J (2010) Lexical analysis of US political speeches. J Quant Linguist 17(2):123–141
Sebastiani F (2002) Machine learning in automatic text categorization. ACM Comput Surv 14(1):1–27
Stamatatos J (2009) A survey of modern authorship attribution methods. J Am Soc Inf Sci Technol 60(3):538–556
Toutanova K, Klein D, Manning C, Singer Y (2003) Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of HLT-NAACL 2003, pp 252–259
Weiss SM, Indurkhya N, Zhang T (2010) Fundamentals of predictive text mining. Springer, London
Yang Y, Pedersen JO (1997) A comparative study of feature selection in text categorization. In: Proceedings of the fourteenth conference on machine learning ICML, pp 412–420
Zheng R, Li J, Chen H, Huang Z (2006) A framework for authorship identification of online messages: writing-style features and classification techniques. J Am Soc Inf Sci Technol 57(3):378–393