A study of Information Retrieval weighting schemes for sentiment analysis

Georgios Paltoglou
University of Wolverhampton
Wolverhampton, United Kingdom
g.paltoglou@wlv.ac.uk

Mike Thelwall
University of Wolverhampton
Wolverhampton, United Kingdom
m.thelwall@wlv.ac.uk

Abstract

Most sentiment analysis approaches use as baseline a support vector machines (SVM) classifier with binary unigram weights. In this paper, we explore whether more sophisticated feature weighting schemes from Information Retrieval can enhance classification accuracy. We show that variants of the classic $tf.idf$ scheme adapted to sentiment analysis provide significant increases in accuracy, especially when using a sublinear function for term frequency weights and document frequency smoothing. The techniques are tested on a wide selection of data sets and produce the best accuracy to our knowledge.

1 Introduction

The increase of user-generated content on the web in the form of reviews, blogs, social networks, tweets, fora, etc. has resulted in an environment where everyone can publicly express their opinion about events, products or people. This wealth of information is potentially of vital importance to institutions and companies, providing them with ways to research their consumers, manage their reputations and identify new opportunities. Wright (2009) claims that "for many businesses, online opinion has turned into a kind of virtual currency that can make or break a product in the marketplace".

Sentiment analysis, also known as opinion mining, provides mechanisms and techniques through which this vast amount of information can be processed and harnessed. Research in the field has mainly, but not exclusively, focused in two subproblems: detecting whether a segment of text, either a whole document or a sentence, is subjective or objective, i.e. contains an expression of opinion, and detecting the overall polarity of the text, i.e. positive or negative.

Most of the work in sentiment analysis has focused on supervised learning techniques (Sebastiani, 2002), although there are some notable exceptions (Turney, 2002; Lin and He, 2009). Previous research has shown that in general the performance of the former tend to be superior to that of the latter (Mullen and Collier, 2004; Lin and He, 2009). One of the main issues for supervised approaches has been the representation of documents. Usually a *bag of words* representation is adopted, according to which a document is modeled as an unordered collection of the words that it contains. Early research by Pang et al. (2002) in sentiment analysis showed that a binary unigram-based representation of documents, according to which a document is modeled only by the presence or absence of words, provides the best baseline classification accuracy in sentiment analysis in comparison to other more intricate representations using bigrams, adjectives, etc.

Later research has focused on extending the document representation with more complex features such as structural or syntactic information (Wilson et al., 2005), favorability measures from diverse sources (Mullen and Collier, 2004), implicit syntactic indicators (Greene and Resnik, 2009), stylistic and syntactic feature selection (Abbasi et al., 2008), "annotator rationales" (Zaidan et al., 2007) and others, but no systematic study has been presented exploring the benefits of employing more sophisticated models for assigning weights to word features.

In this paper, we examine whether term weighting functions adopted from Information Retrieval (IR) based on the standard $tf.idf$ formula and adapted to the particular setting of sentiment analysis can help classification accuracy. We demonstrate that variants of the original $tf.idf$ weighting scheme provide significant increases in classification performance. The advantages of the approach are that it is intuitive, computationally efficient

Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 1386–1395,
Uppsala, Sweden, 11-16 July 2010.

and doesn't require additional human annotation or external sources. Experiments conducted on a number of publicly available data sets improve on the previous state-of-the art.

The next section provides an overview of relevant work in sentiment analysis. In section 3 we provide a brief overview of the original $tf.idf$ weighting scheme along with a number of variants and show how they can be applied to a classification scenario. Section 4 describes the corpora that were used to test the proposed weighting schemes and section 5 discusses the results. Finally, we conclude and propose future work in section 6.

2 Prior Work

Sentiment analysis has been a popular research topic in recent years. Most of the work has focused on analyzing the content of movie or general product reviews, but there are also applications to other domains such as debates (Thomas et al., 2006; Lin et al., 2006), news (Devitt and Ahmad, 2007) and blogs (Ounis et al., 2008; Mishne, 2005). The book of Pang and Lee (2008) presents a thorough overview of the research in the field. This section presents the most relevant work.

Pang et al. (2002) conducted early polarity classification of reviews using supervised approaches. They employed Support Vector Machines (SVMs), Naive Bayes and Maximum Entropy classifiers using a diverse set of features, such as unigrams, bigrams, binary and term frequency feature weights and others. They concluded that sentiment classification is more difficult that standard topic-based classification and that using a SVM classifier with binary unigram-based features produces the best results.

A subsequent innovation was the detection and removal of the objective parts of documents and the application of a polarity classifier on the rest (Pang and Lee, 2004). This exploited text coherence with adjacent text spans which were assumed to belong to the same subjectivity or objectivity class. Documents were represented as graphs with sentences as nodes and association scores between them as edges. Two additional nodes represented the subjective and objective poles. The weights between the nodes were calculated using three different, heuristic decaying functions. Finding a partition that minimized a cost function separated the objective from the subjective sentences. They reported a statistically significant improvement over a Naive Bayes baseline using the whole text but only slight increase compared to using a SVM classifier on the entire document.

Mullen and Collier (2004) used SVMs and expanded the feature set for representing documents with favorability measures from a variety of diverse sources. They introduced features based on Osgood's Theory of Semantic Differentiation (Osgood, 1967) using WordNet to derive the values of potency, activity and evaluative of adjectives and Turney's semantic orientation (Turney, 2002). Their results showed that using a *hybrid* SVM classifier, that uses as features the distance of documents from the separating hyperplane, with all the above features produces the best results.

Whitelaw et al. (2005) added fine-grained semantic distinctions in the feature set. Their approach was based on a lexicon created in a semi-supervised fashion and then manually refined It consists of 1329 adjectives and their modifiers categorized under several taxonomies of appraisal attributes based on Martin and White's Appraisal Theory (2005). They combined the produced appraisal groups with unigram-based document representations as features to a Support Vector Machine classifier (Witten and Frank, 1999), resulting in significant increases in accuracy.

Zaidan et al. (2007) introduced "annotator rationales", i.e. words or phrases that explain the polarity of the document according to human annotators. By deleting rationale text spans from the original documents they created several *contrast* documents and constrained the SVM classifier to classify them less confidently than the originals. Using the largest training set size, their approach significantly increased the accuracy on a standard data set (see section 4).

Prabowo and Thelwall (2009) proposed a *hybrid* classification process by combining in sequence several ruled-based classifiers with a SVM classifier. The former were based on the General Inquirer lexicon (Wilson et al., 2005), the MontyLingua part-of-speech tagger (Liu, 2004) and co-occurrence statistics of words with a set of predefined reference words. Their experiments showed that combining multiple classifiers can result in better effectiveness than any individual classifier, especially when sufficient training data isn't available.

In contrast to machine learning approaches that require labeled corpora for training, Lin and

He (2009) proposed an unsupervised probabilistic modeling framework, based on Latent Dirichlet Allocation (LDA). The approach assumes that documents are a mixture of topics, i.e. probability distribution of words, according to which each document is generated through an hierarchical process and adds an extra sentiment layer to accommodate the opinionated nature (positive or negative) of the document. Their best attained performance, using a filtered subjectivity lexicon and removing objective sentences in a manner similar to Pang and Lee (2004), is only slightly lower than that of a fully-supervised approach.

3 A study of non-binary weights

We use the terms "features", "words" and "terms" interchangeably in this paper, since we mainly focus on unigrams. The approach nonetheless can easily be extended to higher order n-grams. Each document D therefore is represented as a bag-of-words feature vector: $D = \{w_1, w_2, ..., w_{|V|}\}$ where $|V|$ is the size of the vocabulary (i.e. the number of unique words) and w_i, $i = 1, \ldots, |V|$ is the weight of term i in document D.

Despite the significant attention that sentiment analysis has received in recent years, the best accuracy without using complex features (Mullen and Collier, 2004; Whitelaw et al., 2005) or additional human annotations (Zaidan et al., 2007) is achieved by employing a binary weighting scheme (Pang et al., 2002), where $w_i = 1, \text{if } tf_i > 0$ and $w_i = 0, \text{if } tf_i = 0$, where tf_i is the number of times that term i appears in document D (henceforth *raw term frequency*) and utilizing a SVM classifier. It is of particular interest that using tf_i in the document representation usually results in decreased accuracy, a result that appears to be in contrast with topic classification (Mccallum and Nigam, 1998; Pang et al., 2002).

In this paper, we also utilize SVMs but our study is centered on whether more sophisticated than binary or raw term frequency weighting functions can improve classification accuracy. We base our approach on the classic $tf.idf$ weighting scheme from Information Retrieval (IR) and adapt it to the domain of sentiment classification.

3.1 The classic tf.idf weighting schemes

The classic $tf.idf$ formula assigns weight w_i to term i in document D as:

$$w_i = tf_i \cdot idf_i = tf_i \cdot log\frac{N}{df_i} \quad (1)$$

where tf_i is the number of times term i occurs in D, idf_i is the *inverse document frequency* of term i, N is the total number of documents and df_i is the number of documents that contain term i.

The utilization of tf_i in classification is rather straightforward and intuitive but, as previously discussed, usually results in decreased accuracy in sentiment analysis. On the other hand, using idf to assign weights to features is less intuitive, since it only provides information about the general distribution of term i amongst documents of all classes, without providing any additional evidence of class preference. The utilization of idf in information retrieval is based on its ability to distinguish between content-bearing words (words with some semantical meaning) and simple function words, but this behavior is at least ambiguous in classification.

Table 1: SMART notation for *term frequency* variants. $max_t(tf)$ is the maximum frequency of any term in the document and avg_dl is the average number of terms in all the documents. For ease of reference, we also include the BM25 tf scheme. The k_1 and b parameters of BM25 are set to their default values of 1.2 and 0.95 respectively (Jones et al., 2000).

Notation	Term frequency
n (natural)	tf
l (logarithm)	$1 + log(tf)$
a (augmented)	$0.5 + \frac{0.5 \cdot tf}{max_t(tf)}$
b (boolean)	$\begin{cases} 1, & tf > 0 \\ 0, & otherwise \end{cases}$
L (log ave)	$\frac{1+log(tf)}{1+log(avg\_dl)}$
o (BM25)	$\frac{(k_1+1) \cdot tf}{k_1\left((1-b)+b \cdot \frac{dl}{avg\_dl}\right)+tf}$

3.2 Delta tf.idf

Martineau and Finin (2009) provide a solution to the above issue of idf utilization in a classification scenario by localizing the estimation of idf to the documents of one or the other class and subtracting the two values. Therefore, the weight of term

Table 2: SMART notation for *inverse document frequency* variants. For ease of reference we also include the BM25 idf factor and also present the extensions of the original formulations with their Δ variants.

Notation	Inverse Document Frequency
n (no)	1
t (idf)	$log\frac{N}{df}$
p (prob idf)	$log\frac{N-df}{df}$
k (BM25 idf)	$log\frac{N-df+0.5}{df+0.5}$
$\Delta(t)$ (Delta idf)	$log\frac{N_1 \cdot df_2}{N_2 \cdot df_1}$
$\Delta(t')$ (Delta smoothed idf)	$log\frac{N_1 \cdot df_2+0.5}{N_2 \cdot df_1+0.5}$
$\Delta(p)$ (Delta prob idf)	$log\frac{(N_1-df_1)\cdot df_2}{df_1 \cdot (N_2-df_2)}$
$\Delta(p')$ (Delta smoothed prob idf)	$log\frac{(N_1-df_1)\cdot df_2+0.5}{(N_2-df_2)\cdot df_1+0.5}$
$\Delta(k)$ (Delta BM25 idf)	$log\frac{(N_1-df_1+0.5)\cdot df_2+0.5}{(N_2-df_2+0.5)\cdot df_1+0.5}$

i in document D is estimated as:

$$\begin{aligned} w_i &= tf_i \cdot log_2(\frac{N_1}{df_{i,1}}) - tf_i \cdot log_2(\frac{N_2}{df_{i,2}}) \\ &= tf_i \cdot log_2(\frac{N_1 \cdot df_{i,2}}{df_{i,1} \cdot N_2}) \end{aligned} \quad (2)$$

where N_j is the total number of training documents in class c_j and $df_{i,j}$ is the number of training documents in class c_j that contain term i. The above weighting scheme was appropriately named $Delta\, tf.idf$.

The produced results (Martineau and Finin, 2009) show that the approach produces better results than the simple tf or binary weighting scheme. Nonetheless, the approach doesn't take into consideration a number of tested notions from IR, such as the non-linearity of term frequency to document relevancy (e.g. Robertson et al. (2004)) according to which, the probability of a document being relevant to a query term is typically sublinear in relation to the number of times a query term appears in the document. Additionally, their approach doesn't provide any sort of smoothing for the $df_{i,j}$ factor and is therefore susceptible to errors in corpora where a term occurs in documents of only one or the other class and therefore $df_{i,j} = 0$.

3.3 SMART and BM25 tf.idf variants

The SMART retrieval system by Salton (1971) is a retrieval system based on the vector space model (Salton and McGill, 1986). Salton and Buckley (1987) provide a number of variants of the $tf.idf$ weighting approach and present the *SMART notation scheme*, according to which each weighting function is defined by triples of letters; the first one denotes the term frequency factor, the second one corresponds to the inverse document frequency function and the last one declares the normalization that is being applied. The upper rows of tables 1, 2 and 3 present the three most commonly used weighting functions for each factor respectively. For example, a binary document representation would be equivalent to $SMART.bnn$[1] or more simply bnn, while a simple raw term frequency based would be notated as nnn or nnc with cosine normalization.

Table 3: SMART normalization.

Notation	Normalization
n (none)	1
c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+...+w_n^2}}$

Significant research has been done in IR on diverse weighting functions and not all versions of SMART notations are consistent (Manning et al., 2008). Zobel and Moffat (1998) provide an exhaustive study but in this paper, due to space constraints, we will follow the concise notation presented by Singhal et al. (1995).

The BM25 weighting scheme (Robertson et al., 1994; Robertson et al., 1996) is a probabilistic model for information retrieval and is one of the most popular and effective algorithms used in information retrieval. For ease of reference, we incorporate the BM25 tf and idf factors into the SMART annotation scheme (last row of table 1 and 4^{th} row of table 2), therefore the weight w_i of term i in document D according to the BM25 scheme is notated as $SMART.okn$ or okn.

Most of the tf weighting functions in SMART and the BM25 model take into consideration the non-linearity of document relevance to term fre-

[1]Typically, a weighting function in the SMART system is defined as a pair of triples, i.e. $ddd.qqq$ where the first triple corresponds to the document representation and the second to the query representation. In the context that the SMART annotation is used here, we will use the prefix $SMART$ for the first part and a triple for the document representation in the second part, i.e. $SMART.ddd$, or more simply ddd.

quency and thus employ tf factors that scale sublinearly in relation to term frequency. Additionally, the BM25 tf variant also incorporates a scaling for the length of the document, taking into consideration that longer documents will by definition have more term occurences[2]. Effective weighting functions is a very active research area in information retrieval and it is outside the scope of this paper to provide an in-depth analysis but significant research can be found in Salton and McGill (1986), Robertson et al. (2004), Manning et al. (2008) or Armstrong et al. (2009) for a more recent study.

3.4 Introducing SMART and BM25 Delta tf.idf variants

We apply the idea of localizing the estimation of idf values to documents of one class but employ more sophisticated term weighting functions adapted from the SMART retrieval system and the BM25 probabilistic model. The resulting idf weighting functions are presented in the lower part of table 2. We extend the original SMART annotation scheme by adding Delta (Δ) variants of the original idf functions and additionally introduce smoothed Delta variants of the *idf* and the *prob idf* factors for completeness and comparative reasons, noted by their *accented* counterparts. For example, the weight of term i in document D according to the $o\Delta(k)n$ weighting scheme where we employ the BM25 tf weighting function and utilize the difference of class-based BM25 idf values would be calculated as:

$$\begin{aligned} w_i &= \frac{(k_1+1)\cdot tf_i}{K+tf_i}\cdot log(\frac{N_1-df_{i,1}+0.5}{df_{i,1}+0.5}) \\ &- \frac{(k_1+1)\cdot tf_i}{K+tf_i}\cdot log(\frac{N_2-df_{i,2}+0.5}{df_{i,2}+0.5}) \\ &= \frac{(k_1+1)\cdot tf_i}{K+tf_i} \\ &\cdot \quad log\left(\frac{(N_1-df_{i,1}+0.5)\cdot(df_{i,2}+0.5)}{(N_2-df_{i,2}+0.5)\cdot(df_{i,1}+0.5)}\right) \end{aligned}$$

where K is defined as $k_1\left((1-b)+b\cdot\frac{dl}{avg\_dl}\right)$. However, we used a minor variation of the above formulation for all the final *accented* weighting functions in which the smoothing factor is added to the product of df_i with N_i (or its variation for $\Delta(p')$ and $\Delta(k)$), rather than to the df_i alone as the above formulation would imply (see table 2). The above variation was made for two reasons: firstly, when the df_i's are larger than 1 then the smoothing factor influences the final idf value only in a minor way in the revised formulation, since it is added only after the multiplication of the df_i with N_i (or its variation). Secondly, when $df_i = 0$, then the smoothing factor correctly adds only a small mass, avoiding a potential division by zero, where otherwise it would add a much greater mass, because it would be multiplied by N_i.

According to this annotation scheme therefore, the original approach by Martineau and Finin (2009) can be represented as $n\Delta(t)n$.

We hypothesize that the utilization of sophisticated term weighting functions that have proved effective in information retrieval, thus providing an indication that they appropriately model the distinctive power of terms to documents and the smoothed, localized estimation of idf values will prove beneficial in sentiment classification.

Table 4: Reported accuracies on the Movie Review data set. Only the best reported accuracy for each approach is presented, measured by 10-fold cross validation. The list is not exhaustive and because of differences in training/testing data splits the results are not directly comparable. It is produced here only for reference.

Approach	Acc.
SVM with unigrams & binary weights (Pang et al., 2002), reported at (Pang and Lee, 2004)	87.15%
Hybrid SVM with Turney/Osgood Lemmas (Mullen and Collier, 2004)	86%
SVM with min-cuts (Pang and Lee, 2004)	87.2%
SVM with appraisal groups (Whitelaw et al., 2005)	90.2%
SVM with log likehood ratio feature selection (Aue and Gamon, 2005)	90.45%
SVM with annotator rationales (Zaidan et al., 2007)	92.2%
LDA with filtered lexicon, subjectivity detection (Lin and He, 2009)	84.6%

The approach is straightforward, intuitive, computationally efficient, doesn't require additional human effort and takes into consideration standardized and tested notions from IR. The results presented in section 5 show that a number

[2]We deliberately didn't extract the normalization component from the BM25 tf variant, as that would unnecessarily complicate the notation.

of weighting functions solidly outperform other state-of-the-art approaches. In the next section, we present the corpora that were used to study the effectiveness of different weighting schemes.

4 Experimental setup

We have experimented with a number of publicly available data sets.

The movie review dataset by Pang et al. (2002) has been used extensively in the past by a number of researchers (see Table 4), presenting the opportunity to compare the produced results with previous approaches. The dataset comprises 2,000 movie reviews, equally divided between positive and negative, extracted from the Internet Movie Database[3] archive of the *rec.arts.movies.reviews* newsgroup. In order to avoid reviewer bias, only 20 reviews per author were kept, resulting in a total of 312 reviewers[4]. The best attained accuracies by previous research on the specific data are presented in table 4. We do not claim that those results are directly comparable to ours, because of potential subtle differences in tokenization, classifier implementations etc, but we present them here for reference.

The Multi-Domain Sentiment data set (MDSD) by Blitzer et al. (2007) contains Amazon reviews for four different product types: books, electronics, DVDs and kitchen appliances. Reviews with ratings of 3 or higher, on a 5-scale system, were labeled as positive and reviews with a rating less than 3 as negative. The data set contains 1,000 positive and 1,000 negative reviews for each product category for a total of 8,000 reviews. Typically, the data set is used for domain adaptation applications but in our setting we only split the reviews between positive and negative[5].

Lastly, we present results from the BLOGS06 (Macdonald and Ounis, 2006) collection that is comprised of an uncompressed 148GB crawl of approximately 100,000 blogs and their respective RSS feeds. The collection has been used for 3 consecutive years by the Text REtrieval Conferences (TREC)[6]. Participants of the conference are provided with the task of finding documents (i.e. web pages) expressing an opinion about specific entities X, which may be people, companies, films etc. The results are given to human assessors who then judge the content of the webpages (i.e. blog post and comments) and assign each webpage a score: "1" if the document contains relevant, factual information about the entity but no expression of opinion, "2" if the document contains an explicit negative opinion towards the entity and "4" is the document contains an explicit positive opinion towards the entity. We used the produced assessments from all 3 years of the conference in our data set, resulting in 150 different entity searches and, after duplicate removal, 7,930 negative documents (i.e. having an assessment of "2") and 9,968 positive documents (i.e. having an assessment of "4"), which were used as the "gold standard" [7]. Documents are annotated at the document-level, rather than at the post level, making this data set somewhat noisy. Additionally, the data set is particularly large compared to the other ones, making classification especially challenging and interesting. More information about all data sets can be found at table 5.

We have kept the pre-processing of the documents to a minimum. Thus, we have lower-cased all words and removed all punctuation but we have not removed stop words or applied stemming. We have also refrained from removing words with low or high occurrence. Additionally, for the BLOGS06 data set, we have removed all html formatting.

We utilize the implementation of a support vector classifier from the *LIBLINEAR* library (Fan et al., 2008). We use a linear kernel and default parameters. All results are based on leave-one out cross validation accuracy. The reason for this choice of cross-validation setting, instead of the most standard ten-fold, is that all of the proposed approaches that use some form of idf utilize the training documents for extracting document frequency statistics, therefore more information is available to them in this experimental setting.

Because of the high number of possible combinations between tf and idf variants ($6 \cdot 9 \cdot 2 = 108$) and due to space constraints we only present results from a subset of the most representative combinations. Generally, we'll use the cosine normalized variants of unsmoothed delta weighting schemes, since they perform better than their un-

[3] http://www.imdb.com

[4] The dataset can be found at: http://www.cs.cornell.edu/People/pabo/movie-review-data/review_polarity.tar.gz.

[5] The data set can be found at http://www.cs.jhu.edu/mdredze/datasets/sentiment/

[6] http://www.trec.nist.gov

[7] More information about the data set, as well as information on how it can be obtained can be found at: http://ir.dcs.gla.ac.uk/test_collections/blogs06info.html

Table 5: Statistics about the data sets used.

Data set	#Documents	#Terms	#Unique Terms	Average #Terms per Document
Movie Reviews	2,000	1,336,883	39,399	668
Multi-Domain Sentiment Dataset (MDSD)	8,000	1,741,085	455,943	217
BLOGS06	17,898	51,252,850	367,899	2,832

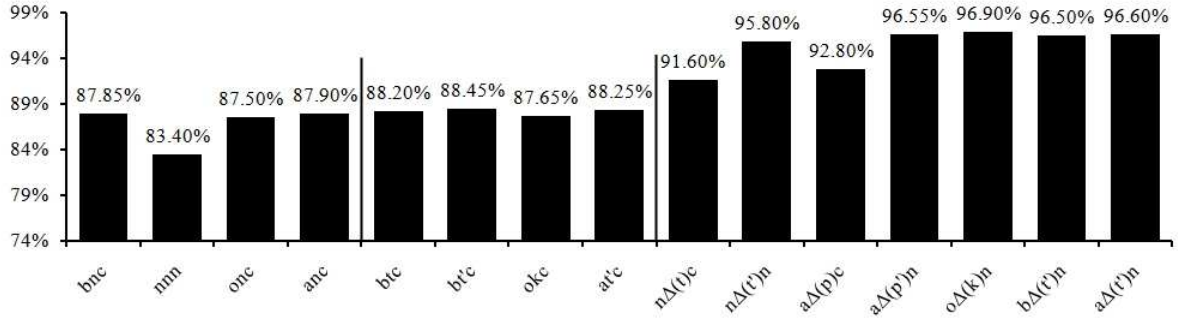

Figure 1: Reported accuracy on the Movie Review data set.

normalized counterparts. We'll avoid using normalization for the smoothed versions, in order to focus our attention on the results of smoothing, rather than normalization.

5 Results

Results for the Movie Reviews, Multi-Domain Sentiment Dataset and BLOGS06 corpora are reported in figures 1, 2 and 3 respectively.

On the Movie Review data set, the results reconfirm that using binary features (bnc) is better than raw term frequency (nnc) (83.40%) features. For reference, in this setting the unnormalized vector using the raw tf approach (nnn) performs similar to the normalized (nnc) (83.40% vs. 83.60%), the former not present in the graph. Nonetheless, using any scaled tf weighting function (anc or onc) performs as well as the binary approach (87.90% and 87.50% respectively). Of interest is the fact that although the BM25 tf algorithm has proved much more successful in IR, the same doesn't apply in this setting and its accuracy is similar to the simpler *augmented tf* approach.

Incorporating un-localized variants of idf (middle graph section) produces only small increases in accuracy. Smoothing also doesn't provide any particular advantage, e.g. btc (88.20%) vs. $bt'c$ (88.45%), since no zero idf values are present. Again, using more sophisticated tf functions provides an advantage over raw tf, e.g. $nt'c$ attains an accuracy of 86.6% in comparison to $at'c$'s 88.25%, although the simpler $at'c$ is again as effective than the BM25 tf ($ot'c$), which performs at 88%. The actual idf weighting function is of some importance, e.g. $ot'c$ (88%) vs. okc (87.65%) and akc (88%) vs. $at'c$ (88.25%), with simpler idf factors performing similarly, although slightly better than BM25.

Introducing smoothed, localized variants of idf and scaled or binary tf weighting schemes produces significant advantages. In this setting, smoothing plays a role, e.g. $n\Delta(t)c$[8] (91.60%) vs. $n\Delta(t')n$ (95.80%) and $a\Delta(p)c$ (92.80%) vs. $a\Delta(p')n$ (96.55%), since we can expect zero class-based estimations of idf values, supporting our initial hypothesis on its importance. Additionally, using *augmented*, BM25 or binary tf weights is always better than raw term frequency, providing further support on the advantages of using sublinear tf weighting functions[9]. In this setting, the best accuracy of 96.90% is attained using BM25 tf weights with the BM25 delta idf variant, although binary or *augmented* tf weights using

[8]The original *Delta tf.idf* by Martineau and Finin (2009) has a limitation of utilizing features with $df > 2$. In our experiments it performed similarly to $n\Delta(t)n$ (90.60%) but still lower than the $cosine$ normalized variant $n\Delta(t)c$ included in the graph (91.60%).

[9]Although not present in the graph, for completeness reasons it should be noted that $l\Delta(s)n$ and $L\Delta(s)n$ also perform very well, both reaching accuracies of approx. 96%.

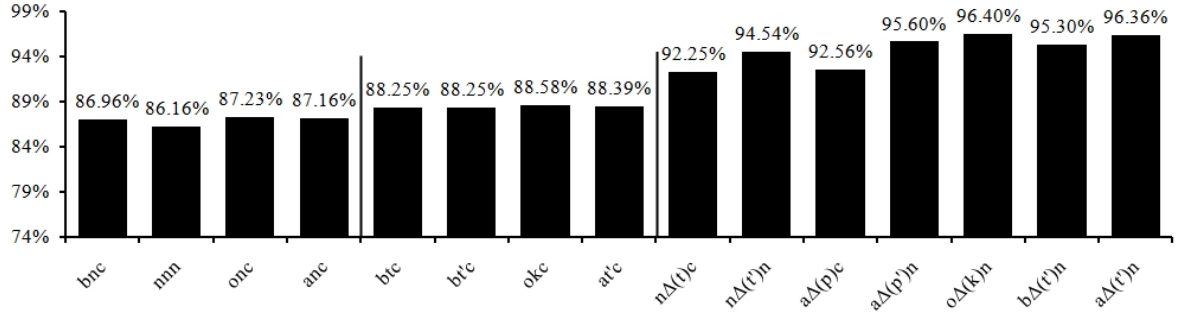

Figure 2: Reported accuracy on the Multi-Domain Sentiment data set.

delta idf perform similarly (96.50% and 96.60% respectively). The results indicate that the tf and the idf factor themselves aren't of significant importance, as long as the former are scaled and the latter smoothed in some manner. For example, $a\Delta(p')n$ vs. $a\Delta(t')n$ perform quite similarly.

The results from the Multi-Domain Sentiment data set (figure 2) largely agree with the findings on the Movie Review data set, providing a strong indication that the approach isn't limited to a specific domain. Binary weights outperform *raw term frequency* weights and perform similarly with scaled tf's. Non-localized variants of idf weights do provide a small advantage in this data set although the actual idf variant isn't important, e.g. btc, $bt'c$, and okc all perform similarly. The utilized tf variant also isn't important, e.g. $at'c$ (88.39%) vs. $bt'c$ (88.25%).

We focus our attention on the $delta\ idf$ variants which provide the more interesting results. The importance of smoothing becomes apparent when comparing the accuracy of $a\Delta(p)c$ and its smoothed variant $a\Delta(p')n$ (92.56% vs. 95.6%). Apart from that, all smoothed $delta\ idf$ variants perform very well in this data set, including somewhat surprisingly, $n\Delta(t')n$ which uses raw tf (94.54%). Considering that the average tf per document is approx. 1.9 in the Movie Review data set and 1.1 in the MDSD, the results can be attributed to the fact that words tend to typically appear only once per document in the latter, therefore minimizing the difference of the weights attributed by different tf functions[10]. The best attained accuracy is 96.40% but as the MDSD has mainly been used for domain adaptation applications, there is no clear baseline to compare it with.

[10]For reference, the average tf per document in the BLOGS06 data set is 2.4.

Lastly, we present results on the BLOGS06 dataset in figure 3. As previously noted, this data set is particularly noisy, because it has been annotated at the document-level rather than the post-level and as a result, the differences aren't as profound as in the previous corpora, although they do follow the same patterns. Focusing on the delta idf variants, the importance of smoothing becomes apparent, e.g. $a\Delta(p)c$ vs. $a\Delta(p')n$ and $n\Delta(t)c$ vs. $n\Delta(t')n$. Additionally, because of the fact that documents tend to be more verbose in this data set, the scaled tf variants also perform better than the simple *raw tf* ones, $n\Delta(t')n$ vs. $a\Delta(t')n$. Lastly, as previously, the smoothed localized idf variants perform better than their unsmoothed counterparts, e.g. $n\Delta(t)n$ vs. $n\Delta(t')n$ and $a\Delta(p)c$ vs. $a\Delta(p')n$.

6 Conclusions

In this paper, we presented a study of document representations for sentiment analysis using term weighting functions adopted from information retrieval and adapted to classification. The proposed weighting schemes were tested on a number of publicly available datasets and a number of them repeatedly demonstrated significant increases in accuracy compared to other state-of-the-art approaches. We demonstrated that for accurate classification it is important to use term weighting functions that scale sublinearly in relation to the number of times a term occurs in a document and that document frequency smoothing is a significant factor.

In the future we plan to test the proposed weighting functions in other domains such as topic classification and additionally extend the approach to accommodate multi-class classification.

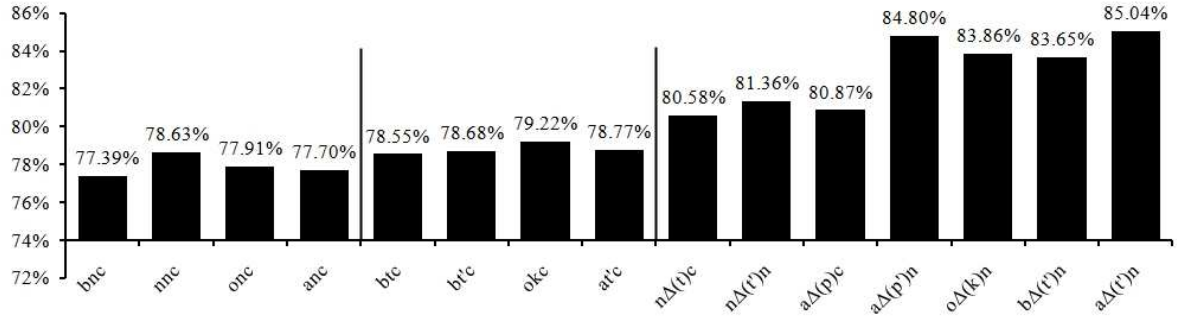

Figure 3: Reported accuracy on the BLOGS06 data set.

Acknowledgments

This work was supported by a European Union grant by the 7th Framework Programme, Theme 3: Science of complex systems for socially intelligent ICT. It is part of the CyberEmotions Project (Contract 231323).

References

Ahmed Abbasi, Hsinchun Chen, and Arab Salem. 2008. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Trans. Inf. Syst.*, 26(3):1–34.

Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009. Improvements that don't add up: ad-hoc retrieval results since 1998. In David Wai Lok Cheung, Il Y. Song, Wesley W. Chu, Xiaohua Hu, Jimmy J. Lin, David Wai Lok Cheung, Il Y. Song, Wesley W. Chu, Xiaohua Hu, and Jimmy J. Lin, editors, *CIKM*, pages 601–610, New York, NY, USA. ACM.

Anthony Aue and Michael Gamon. 2005. Customizing sentiment classifiers to new domains: A case study. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, Prague, Czech Republic, June. Association for Computational Linguistics.

Ann Devitt and Khurshid Ahmad. 2007. Sentiment polarity identification in financial news: A cohesion-based approach. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 984–991, Prague, Czech Republic, June. Association for Computational Linguistics.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Stephan Greene and Philip Resnik. 2009. More than words: Syntactic packaging and implicit sentiment. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 503–511, Boulder, Colorado, June. Association for Computational Linguistics.

K. Sparck Jones, S. Walker, and S. E. Robertson. 2000. A probabilistic model of information retrieval: development and comparative experiments. *Inf. Process. Manage.*, 36(6):779–808.

Chenghua Lin and Yulan He. 2009. Joint sentiment/topic model for sentiment analysis. In *CIKM '09: Proceeding of the 18th ACM conference on Information and knowledge management*, pages 375–384, New York, NY, USA. ACM.

Wei-Hao Lin, Theresa Wilson, Janyce Wiebe, and Alexander Hauptmann. 2006. Which side are you on? identifying perspectives at the document and sentence levels. In *Proceedings of the Conference on Natural Language Learning (CoNLL)*.

Hugo Liu. 2004. MontyLingua: An end-to-end natural language processor with common sense. Technical report, MIT.

C. Macdonald and I. Ounis. 2006. The trec blogs06 collection : Creating and analysing a blog test collection. *DCS Technical Report Series*.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition, July.

J. R. Martin and P. R. R. White. 2005. *The language of evaluation : appraisal in English / J.R. Martin and P.R.R. White*. Palgrave Macmillan, Basingstoke :.

Justin Martineau and Tim Finin. 2009. Delta TFIDF: An Improved Feature Space for Sentiment Analysis. In *Proceedings of the Third AAAI Internatonal Conference on Weblogs and Social Media*, San Jose, CA, May. AAAI Press. (poster paper).

A. Mccallum and K. Nigam. 1998. A comparison of event models for naive bayes text classification.

G. Mishne. 2005. Experiments with mood classification in blog posts. In *1st Workshop on Stylistic Analysis Of Text For Information Access*.

Tony Mullen and Nigel Collier. 2004. Sentiment analysis using support vector machines with diverse information sources. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 412–418, Barcelona, Spain, July. Association for Computational Linguistics.

Charles E. Osgood. 1967. *The measurement of meaning / [by] [Charles E. Osgood, George J. Suci [and] Percy H. Tannenbaum]*. University of Illinois Press, Urbana :, 2nd ed. edition.

Iadh Ounis, Craig Macdonald, and Ian Soboroff. 2008. Overview of the trec-2008 blog trac. In *The Seventeenth Text REtrieval Conference (TREC 2008) Proceedings*. NIST.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *In Proceedings of the ACL*, pages 271–278.

B. Pang and L. Lee. 2008. *Opinion Mining and Sentiment Analysis*. Now Publishers Inc.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Rudy Prabowo and Mike Thelwall. 2009. Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157, April.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at trec-3. In *TREC*, pages 0–.

S E Robertson, S Walker, S Jones, M M Hancock-Beaulieu, and M Gatford. 1996. Okapi at trec-2. In *In The Second Text REtrieval Conference (TREC-2), NIST Special Special Publication 500-215*, pages 21–34.

Stephen Robertson, Hugo Zaragoza, and Michael Taylor. 2004. Simple bm25 extension to multiple weighted fields. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 42–49, New York, NY, USA. ACM.

Gerard Salton and Chris Buckley. 1987. Term weighting approaches in automatic text retrieval. Technical report, Ithaca, NY, USA.

Gerard Salton and Michael J. McGill. 1986. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.

G. Salton. 1971. *The SMART Retrieval System—Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1ñ47.

Amit Singhal, Gerard Salton, and Chris Buckley. 1995. Length normalization in degraded text collections. Technical report, Ithaca, NY, USA.

Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. *CoRR*, abs/cs/0607062.

Peter D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424.

Casey Whitelaw, Navendu Garg, and Shlomo Argamon. 2005. Using appraisal groups for sentiment analysis. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 625–631, New York, NY, USA. ACM.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, Vancouver, CA.

Ian H. Witten and Eibe Frank. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (The Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, 1st edition, October.

Alex Wright. 2009. Mining the web for feelings, not facts. August 23, NY Times, last accessed October 2, 2009, http://http://www.nytimes.com/2009/08/24/technology/internet/ 24emotion.html?_r=1.

O.F. Zaidan, J. Eisner, and C.D. Piatko. 2007. Using Annotator Rationales to Improve Machine Learning for Text Categorization. *Proceedings of NAACL HLT*, pages 260–267.

Justin Zobel and Alistair Moffat. 1998. Exploring the similarity space. *SIGIR Forum*, 32(1):18–34.