Sentiment Analysis of Textual Reviews

Evaluating Machine Learning, Unsupervised and SentiWordNet Approaches

V.K. Singh, R. Piryani, A. Uddin
Department of Computer Science
South Asian University
New Delhi, India
vivek@cs.sau.ac.in, rajesh.piryani@gmail.com,
mdaakib18@gmail.com

P. Waila, Marisha
DST Centre for Interdisciplinary Mathematical Sciences,
Banaras Hindu University
Varanasi, India
pranav.waila@gmail.com, shangu90@gmail.com

Abstract—**This paper presents our experimental results on performance evaluation of all the three approaches for document-level sentiment classification. We have implemented two Machine Learning based classifiers (Naïve Bayes and SVM), the Unsupervised Semantic Orientation approach (SO-PMI-IR algorithm) and the SentiWordNet approaches for sentiment classification of movie reviews. We used two pre-existing large datasets and collected one of moderate size on our own. The paper primarily makes two useful contributions: (a) it presents a comprehensive evaluative account of performance of all the three available approaches on use with movie reviews, and (b) it presents a new modified Adjective+Adverb combine scheme of SentiWordNet approach.**

Keywords: Sentiment Analysis, Naïve Bayes, Support Vector Machine, Semantic Orientation Approach, SentiWordNet.

I. INTRODUCTION

The recent transformation of the World Wide Web into a more participative and co-creative Web has allowed a large number of users to post their contributions on the World Wide Web. Even those who are virtually novice to the technicalities of the Web publishing are creating content on the World Wide Web. This ease of content creation is seen in a variety of forms such as writing blogs, writing in forums, writing reviews in an easy to use form, rating items & services displayed on the Web, uploading pictures and videos etc. In fact the value of a Website is now determined largely by its user base and the data available on it. Any one may create a website similar in technical functionality to Amazon, Youtube or facebook. But it is extremely difficult to achieve the equivalent traffic. It may perhaps be true to say that Data is the new Intel inside.

The tremendous amount of information available on the World Wide Web in form of user-reviews for different items ranging from mobile phones, holiday trips, hotel services to movie reviews etc. is now a valuable knowledge source. For example, a user looking for a hotel in a particular tourist city may see the reviews of available hotels in the city while making a decision to book in one of them. Or a user willing to buy a particular model of computer printer may look at reviews posted by many other users about that printer before making a buying decision.This not only helps in locating the relevant and useful information in time but also in arriving at a more informed decision. Textual reviews are one such interesting domain with great amount of possible usage and applications.

With the ease of content creation on the Web, now users are posting reviews on various sites. For example, if one browses a website like *carwale.com,* it is natural to find it full of reviews of lot of users. These users are generally of different kinds. For example for a particular car available in India since last 10 years, one may find reviews written by users who are using it for 8 years, or 5 years or so, or those who have either recently brought it or planning to buy it. Similarly, a look at the Internet movie database website (*www.imdb.com*) will show that it is full of large number of user reviews for virtually exhaustive set of movies produced and released in the World.

Though these reviews are beyond doubt very useful and valuable, but at the same time it is also quite difficult for a new user (or a prospective customer) to read all the reviews in a short span of time. Fortunately we have a solution to this information overload problem which can present a comprehensive summary result out of a large number of reviews. The new IR techniques now not only allow to automatically label a review as positive or negative, but they also allow extracting and highlighting positive and negative aspects of a product/ service. This particular IR task is formally known as Sentiment Analysis or Opinion Mining.

Sentiment analysis is now an important part of IR based formulations in a variety of domains. It is traditionally used for automatic extraction of opinions types about a product and for highlighting positive or negative aspects/ features of a product. Recently we have seen use of sentiment analysis for opinion based clustering of text-documents and for providing better and more focused recommendations by a recommender system [1], [2]. The document-level sentiment analysis problem is essentially as follows: Given a set of documents D, a sentiment analysis algorithm classifies each document $d \epsilon D$ into one of the two classes, *positive* and *negative*. Positive label denotes that the document d expresses a positive opinion and negative label means that d expresses a negative opinion of the user. Sometimes a *neutral* class is used as

978-1-4673-4853-9/13/$31.00 ©2013 IEEE

Authorized licensed use limited to: b-on: UNIVERSIDADE DE AVEIRO. Downloaded on March 19,2025 at 19:55:39 UTC from IEEE Xplore. Restrictions apply.

well to label those documents which are largely narrative or explanatory and do not express any subjective opinion.

There are primarily three types of approaches for sentiment classification of opinionated texts: (a) using a machine learning based text classifier -such as Naïve Bayes, SVM or kNN- with suitable feature selection scheme; (b) using Semantic Orientation scheme of extracting relevant n-grams of the text and then labeling them either as positive or negative and consequentially the document; and (c) using the SentiWordNet based publicly available library that provides positive, negative and neutral scores for words. Some of the relevant past works on sentiment classification can be found in [3], [4], [5], [6], [7], [8], [9], &[10].

This paper primarily makes two useful contributions to the knowledge in this area: (a) it presents a comprehensive evaluative account of performance of all the available techniques on use with movie reviews, and (b) it presents a new Adjective+Adverb combine scheme of SentiWordNet approach. The rest of the paper is organized as follows. Section II of the paper briefly describes the Naïve Bayes (NB) and Support Vector Machine (SVM) classifiers. Section III describes the unsupervised semantic orientation approach and Section IV describes the SentiWordNet approach. Section V explains the experimental setup and datasets used and Section VI present the results obtained on various performance benchmarks. The paper concludes with observations and conclusion described in section VII.

II. MACHINE LEARNING BASED CLASSIFIERS

The machine learning based text classifiers are a kind of supervised machine learning paradigm, where the classifier needs to be trained on some labeled training data before it can be applied to actual classification task. The training data is usually an extracted portion of the original data hand labeled manually. After suitable training they can be used on the actual test data. The Naïve Bayes is a statistical classifier whereas SVM is a kind of vector space classifier.

A. *Naïve Bayes Algorithm*

The statistical text classifier scheme of Naïve Bayes (NB) can be adapted to be used for sentiment classification problem as it can be visualized as a 2-class text classification problem. The issue that remains to be addressed next is whether all terms occurring in documents should be used as features as done in normal text classification task or we should select specific terms which may be more concrete forms of expression of opinions. We have used both these variants. For the sake of completeness we are briefly describing the algorithmic approach of NB and its key expressions. A more detailed description can be found in [11] & [12]. It is a probabilistic learning method which computes the probability of a document *d* being in class *c* as in (1) below.

$$P\,(c|d)\ \alpha\ P(c)\prod_{1\le k\le nd} P(t_k|c) \qquad \textbf{(1)}$$

where, P (t_k/c) is the conditional probability of a term t_k occurring in a document of class c. The expression P(t_k/c) is a measure of how much evidence the term t_k contributes that c is correct class. P(c) is the prior probability of a document occurring in class c and usually corresponds to the majority class. The goal in text classification is to find the best class for a document. The key idea in this classification is thus to categorize documents based on statistical pattern of occurrence of terms. The class membership of a document can be computed as in (2).

$$c_{map} = arg \max_{c\in C} \widehat{\mathrm{P}}(c|d) = arg \max_{c\in C} \widehat{\mathrm{P}}(c) \prod_{1\le k\le n_d} \widehat{\mathrm{P}}\,(t_k|c) \qquad \textbf{(2)}$$

where, P` is an estimated value obtained from the training set. In order to reduce the computational complexity resulting from multiplication of large number of probability terms, the eq. (2) can be transformed to (3).

$$C_{map} = arg \max_{c\in C}[\log \hat{P}(c) + \sum_{1\le k\le n_d} \log \hat{P}(t_k|c)] \qquad \textbf{(3)}$$

Each conditional parameter P(t_k/c) in (3) is a weight that indicates how good an indicator the term t_k is for class c, and the prior log P(c) indicates the relative frequency of class c. NB has two popular variants: the multinomial NB and the Bernoulli's NB. In the former we take into account both the presence or absence of a term and also its frequency of occurrence as an indicator for a particular class. On the other hand in Bernoulli's NB we simply take into account only whether a term is present or absent in a document and not how many times it occurs in the document.We have implemented the multinomial NB for our classification task.

B. *Support Vector Machine Algorithm*

Support Vector machine (SVM) is a kind of vector space model based classifier which requires that the text documents should be transformed to feature vectors before they are used for classification. Usually the text documents are transformed to multidimensional *tf.idf* vectors. The entire problem of classification is then classifying every text document represented as a vector into a particular class. It is a type of large margin classifier. Here the goal is to find a decision boundary between two classes that is maximally far from any document in the training data. The key idea of SVM is to find a decision surface that is maximally away from any data point (documents in our case). The distance from the decision surface to the closest data point determines the margin of the classifier. These separator points are referred to as support vectors. Maximizing the margin by SVM reduces the uncertain classification decisions.

If the training set is D = {{x_i,y_i)}, where each member is a pair of point x and corresponding label y, then assuming that two data classes are always named +1 and -1, we can visualize the linear classifier as:

$$\mathrm{f(x) = sign}(w^T x + b) \qquad \textbf{(4)}$$

Here a value of -1 indicates one class, and a value of +1 indicates the other class. The algebraic manipulation then involves defining the functional margin and the geometric margin, and the final standard formulation of SVM becomes a minimization problem that tries to find w and b such that (a) ½ w^Tw is minimized, and (b) for all $\{(x_i,y_i)\}$, $y_i(w^Tx_i+b) >=1$. This is a quadratic optimization problem and can be solved using standard quadratic programming libraries. In this case the solution involves constructing a dual problem where a Lagrange multiplier α_i is associated with each constant y_i $(w^Tx_i+ b) >= 1$ in the primal problem:- Find α_1, α_2, …α_N such that

Authorized licensed use limited to: b-on: UNIVERSIDADE DE AVEIRO. Downloaded on March 19,2025 at 19:55:39 UTC from IEEE Xplore. Restrictions apply.

$$\sum \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i^T x_j \qquad \textbf{(5)}$$

is maximized subject to constraints $\Sigma_i\ \alpha_i y_i = 0$ and $\alpha_i >= 0$ for all 1 <= i<=N. The solution is then of the form:

$$w = \sum \alpha_i y_i x_i$$
$$b = y_k - w^T x_k\ for\ any\ such\ x_k\ s.t. \alpha_k \neq 0 \qquad \textbf{(6)}$$

The classification function thus becomes:

$$f\ (x) = sign\ (\sum_i \alpha_i y_i x_i^T x + b) \qquad \textbf{(7)}$$

where, most of the α_i are zero and each non-zero α_i indicates that the corresponding x_i is a support vector. We have used Sequential Minimal Optimizer (SMO) available in weka [13] to solve this convex quadratic optimization problem as stated above. SMO breaks the quadratic programming problem into many small problems, solving these problems sequentially gives the same answer as solving the big quadratic convex problem.The SVM can thus be used as a sentiment classifier since it is essentially doing a 2-class classification.

III. UNSUPERVISED SEMANTIC ORIENTATION APPROACH

The semantic orientation approach uses an unsupervised scheme of sentiment classification. The sentiment label of a text document here is classified based on aggregate semantic orientation score of selected phrases in it. We first extract phrases that conform to a specific POS [14], [15] and then the semantic orientation of extracted phrases is computed using the Pointwise Mutual Information (PMI) measure given in eq. (8):

$$PMI(term_1, term_2) = \log_2\{\Pr(term_1 \Delta term_2)/\Pr(term_1).\Pr(term_2)\} \qquad \textbf{(8)}$$

where, $\Pr(term_1 \Delta\ term_2)$ is the co-occurrence probability of $term_1$ and $term_2$ and $\Pr(term_1).\Pr(term_2)$ gives the probability that two terms co-occur if they arestatistically independent. The ratio between $\Pr(term_1 \Delta\ term_2)$ and $\Pr(term_1).\Pr(term_2)$ is a measure of the degree of statistical independence between them. The log of this ratio is the amount of information that we acquire about the presence of one word when we observe the other. The Semantic Orientation (SO) of a phrase is now computed by using the eq. (9):

$$SO(phrase) = PMI(pharse, \text{"excellent"}) - PMI(phrase, \text{"}poor\text{"}) \qquad \textbf{(9)}$$

where, PMI (phrase, “excellent”) measures the association of the phrase with positive reference word “excellent” and PMI (phrase, “poor”) measures the association of phrase with negative reference word “poor”. These probabilities are calculated by issuing search query of the form “phrase * excellent” and “phrase * poor” to a search engine. The number of hits obtained is used as a measure of the probability value. The SO value for all the extracted phrases is computed using this scheme. Thereafter the sentiment label of the entire document is assigned based on aggregation of all these scores in the document. One simple aggregation scheme is to assign “+1” score for every positively oriented phrase and “-1” for every negatively oriented phrase. The sentiment label of a document is therefore positive or negative depending on the sum of these scores being positive (or above a threshold) or negative (or below a threshold).

IV. SENTIWORDNET APPROACH

The SentiWordNet approach involves use of the publicly available library of SentiWordNet [16]. In this lexical resource each term *t* occurring in WordNet is associated to three numerical scores obj(t), pos(t) and neg(t), describing the objective, positive and negative polarities of the term, respectively. These three scores are computed by combining the results produced by eight ternary classifiers. To make use of SentiWordNet we need to first extract relevant opinionated terms and then lookup for their scores in the SentiWordNet. Past works in the area have shown that adjectives, adjective+adverb and adjective+verb combinations are few reasonable choices for the terms to be extracted. In English language, adjectives are largely used in an opinionated tone and adverbs are usually used as complements or modifiers. Few examples of adverb usage are: *he ran quickly, only adults, very dangerous trip, very nicely etc*. In all these examples adverbs modify the adjectives. Though adverbs are of various kinds, but for sentiment classification only adjectives of degree seem useful. We have used improved versions of Variable Scoring and Adjective Priority Scoring methods based on SentiWordNet as proposed in [17]. The Variable scoring scheme allows modifying adjective scores and the Adjective priority scoring scheme allows scoring an adjective+adverb combine by assigning a fixed weight to relevance of adverbs.

We have modified the Variable Scoring and Adjective priority scoring methods to simplify them and achieve better accuracy. One simple modification that we introduced is that instead of limiting the scores of adverbs between 0 and 1 and of adjectives between -1 and +1, we take the actual scores obtained from the SentiWordNet. The scoring system is then used to assign different weights to adjectives and adverbs. If we assume ‘adv’ denotes an adverb and ‘adj’ denotes an adjective, then the scoring rules,we used, can be specified as follows:

- If adv is affirmative, then
    - o If score(adj)>0 and score(adv)>0
        - ▪ f_{VS} (adv,adj)= score(adj)+(1-score(adj))*score(adv)
    - o If score(adj)<0 and score(adv)>0
        - ▪ f_{VS} (adv,adj)= score(adj)-(1-score(adj))*score(adv)
- If score(adv) is negative, then
    - o If score(adj)>0 and score(adv)<0
        - ▪ f_{VS} (adv,adj)= score(adj)+(1-score(adj))*score(adv)
    - o If score(adj)<0 and score(adv)<0
        - ▪ f_{VS} (adv,adj)= score(adj)-(1-score(adj))*score(adv)

Here, the scores are the actual SentiWordNet scores. We have also implemented the Adjective Priority Scoring method with slight modification with the value of the scaling factor r = 0.35 (equivalent to giving 35% weight to adverb scores). The modified rules for this method are as follows:

Authorized licensed use limited to: b-on: UNIVERSIDADE DE AVEIRO. Downloaded on March 19,2025 at 19:55:39 UTC from IEEE Xplore. Restrictions apply.

```
• If adv is affirmative, then
    o If score(adj)>0 and score(adv)>0
        ▪ $f_{APS}$ (adv,adj)=
          min(1,score(adj)+r*score(adv))
    o If score(adj)<0 and score(adv)>0
        ▪ $f_{APS}$ (adv,adj)=
          min(1,score(adj)-r*score(adv))
• If score(adv)  is negative, then
    o If score(adj)>0 and score(adv)<0
        ▪ $f_{APS}$(adv,adj)= max(-
          1,score(adj)+r*score(adv))
    o If score(adj)<0 and score(adv)<0
        ▪ $f_{APS}$(adv,adj)= max(-
          1,score(adj)-r*score(adv))
```

We implemented both the scoring models along with weight factor to adjust the proportionate scores. We have used three different schemes for sentiment classification. In one we use only variable Scoring method, in second method we use Adjective Priority Scoring and in the third one we combine both of them. In the first two schemes, a review text is classified as 'positive' or 'negative' based on the fact whether aggregate sum of f values for extracted 'Adj+Adv' combine is positive or negative. However, in the combined scheme a review is classified as 'positive' if it is labeled positive by both the scoring methods, 'negative' if it is labeled negative by both the scoring methods and 'neutral' in cases of conflicting assignments by both schemes.

V. DATASETS AND EXPERIEMNTAL SETUP

We evaluated performance of NB, SVM, SO-PMI-IR and modified SentiWordNet scoring for sentiment analysis on three different movie review data sets. The manual effort involved in this work involves labeling the third data set collected by us and issuing a large number of Google queries for obtaining PMI values for a large number of phrases for use with SO-PMI-IR. The programming environment comprised of JAVA with use of weka APIs for SMO.

A. *Collecting Datasets*

We used two existing standard movie review data sets obtained from Cornell sentiment polarity dataset [18]. We downloaded *polarity dataset v2.0* (referred as dataset 1) and *v1.0* (referred as dataset 2). The *dataset 1* comprises of 1000 positive and 1000 negative processed reviews, whereas the *dataset2* comprises of 700 positive and 700 negative processed reviews. Our third dataset (referred as dataset 3) is our own collection comprising of 1000 reviews of Hindi movies. We obtained 10 reviews each of 100 Hindi movies from the movie database site IMDB [19].

B. *Implementing NB and SVM Algorithms*

We have implemented the multinomial version of NB algorithm using JAVA with Eclipse IDE. All the labeled datasets have been fed to the NB algorithm as k-folds, where k has been chosen to be 3, 5 and 10. These values are an intelligent institutive work which we chose to achieve more confidence in our results. A 3-fold application of test data means that the dataset is divided into three equal parts and then two of the three parts becomes the training data and remaining one part constitute test data. This is done by choosing each of the possible permutations as training data and the corresponding remaining part as test set in different runs. Similarly in 5-fold and 10-fold, the data is divided into 5 and 10 parts respectively. We have taken the entire set of terms as features so as to allow us to compare the results with the adjective+adverb combination of SentiWordNet implementation. We have taken average performance of different runs for each of the k-folds as our final reported result.

The SVM algorithm being a vector space model based classifier required us to first transform the textual movie reviews to vector space representation. We used *tf.idf* representation for transforming the textual reviews to numerical vectors for all the three datasets. No stop word removal or stemming was performed. This was done purposefully so that no feature having sentimental value gets excluded in the representation. We have thereafter used the same fold scheme as stated earlier and run our implementation of SVM and observed the results. The reported results on different performance metrics as stated below are average of multiple runs.

C. *Implementing SO-PMI-IR Algorithm*

The unsupervised SO-PMI-IR algorithm was also implemented in JAVA with use of a POS tagger. First of all the textual reviews were subjected to POS tagging and then once all the reviews were tagged, feature extraction was done for every review. In order to compute the semantic orientation of a review, we extracted all adjectives (by preserving any preceding not) from the review and obtained the SO values of all extracted adjectives from a lookup table that we built separately. This lookup table was built offline by issuing hand coded queries to google and computing the SO values of various phrases. The SO score of a review document was thus an aggregation of all SO values of different adjectives occurring in that document. This aggregation was done as follows: for every positive term (having SO value greater than a threshold, we choose values between 0.8712 to 0.956, as our experimental values) a '+1' score is added to SO of the review document and for every negative term, a '-1' score is added with the SO of the review document.Scores for terms preceded by 'NOT' were complemented. The consolidated SO score of every review document is thus sum of all SO values computed for terms extracted from it. We set a threshold score of +5 for classifying a review as positive. Thus every review is classified either as positive or negative based on the fact whether its aggregate SO score is greater than '+5' or not. This value 5 was set based on observations in multiple trials.

D. *Implementing SentiWordNet scheme*

The SentiWordNet scheme has been implemented on all the three datasets with three scoring methods after performing POS tagging and feature extraction.We first applied POS tagging on the review data and then extracted adjectives along with two words preceding it (assuming possibility of occurrence of adverb and not). We denote the three scoring schemes as SWN(VS), SWN(APS) and SWN(VS+APS). Here we allow neutral label assignments as well. The neutral assignment is rare in first two scoring methods, but is seen in SWN(VS+APS) since there are

Authorized licensed use limited to: b-on: UNIVERSIDADE DE AVEIRO. Downloaded on March 19,2025 at 19:55:39 UTC from IEEE Xplore. Restrictions apply.

many conflicting assignments between the two scoring schemes. The conflicting assignments are labeled ‘neutral’ in our case. While computing scores for adj+adv combine, we have complemented all scores which are preceded by the word ‘not’.

E. *Performance Metrics Computed*

In order to evaluate the accuracy and performance of different algorithms implemented, we computed the standard performance metrics of Accuracy, Precision, Recall and F-measure. Accuracy is measured in percentage, whereas Precision, Recall and F-measure metric values range from 0 – 1. The best accuracy value is 100%, and for the other metric best value will be‘1’.

VI. RESULTS

We obtained sentiment analysis results on three datasets using six different implementations of the four methods evaluated. The table I present the computed results on dataset1 with all the three implementations. The results reported are for 3-fold, 5-fold and 10-fold classification exercise. Similarly tables II and III present corresponding results for dataset 2 and dataset 3, respectively. The term SWN is used as short form of the SentiWordNet algorithm. We have also kept record of total number of reviews assigned ‘positive’ or ‘negative’ by a particular method on all the three datasets. This is presented in table IV.The detailed assignment statistics for the three datasets for three different versions of SentiWordNet scheme is presented in table V. A comprehensive view of accuracy values of all the six implementations on three different datasets is presented in figure 1.

TABLE I. COMPUTED SCORES ON DATASET1 (HAVING 1000 POSITIVE AND 1000 NEGATIVE REVIEWS)

Method	*Performance measure*	*3-fold*	*5-fold*	*10-fold*
NB	Accuracy	83.8%	82.9%	83.35%
	Precision	0.838	0.828	0.834
	Recall	0.838	0.831	0.834
	F-measure	0.838	0.829	0.833
SVM	Accuracy	78.15%	77.95%	79%
	Precision	0.782	0.78	0.79
	Recall	0.782	0.78	0.79
	F-measure	0.781	0.779	0.79
SO-PMI-IR	Accuracy	84.327%		
SWN (VS)	Accuracy	65.15%		
SWN (APS)	Accuracy	65.9%		
SWN (VS+APS)	Accuracy	64.0%		

TABLE II. COMPUTED SCORES ON DATASET2 (HAVING 700 POSITIVE AND 700 NEGATIVE REVIEWS)

Method	*Performance measure*	*3-fold*	*5-fold*	*10-fold*
NB	Accuracy	79.86%	81.07%	81.14%
	Precision	0.799	0.812	0.812
	Recall	0.799	0.811	0.811
	F-measure	0.798	0.811	0.811
SVM	Accuracy	75.64%	76.78%	77.07%
	Precision	0.757	0.768	0.771
	Recall	0.756	0.768	0.771
	F-measure	0.756	0.768	0.771
SO-PMI-IR	Accuracy	82.45%		
SWN (VS)	Accuracy	64.85%		
SWN(APS)	Accuracy	65.21%		
SWN (VS+APS)	Accuracy	63.42%		

TABLE III. COMPUTED SCORES ON DATASET3 (HAVING 10 REVIEWS EACH OF 100 HINDI MOVIES)

Method	*Performance measure*	*3-fold*	*5-fold*	*10-fold*
NB	Accuracy	88.8%	88.7%	89.4%
	Precision	0.886	0.888	0.894
	Recall	0.886	0.887	0.894
	F-measure	0.887	0.888	0.894
SVM	Accuracy	83.3%	84.2%	83.3%
	Precision	0.825	0.838	0.833
	Recall	0.833	0.842	0.838
	F-measure	0.828	0.839	0.835
SO-PMI-IR	Accuracy	89%		
SWN (VS)	Accuracy	66.23%		
SWN (APS)	Accuracy	67.35%		
SWN (VS+APS)	Accuracy	65.12%		

TABLE IV. TOTAL PERCENTAGE OF ‘POSITIVE’ AND ‘NEGATIVE’ LABELS ASSIGNED BY ALL FOUR METHODS (BEST FOLD)

Method	**Dataset**			
		Dataset1	*Dataset2*	*Dataset3*
NB	Positive	49.65%	48%	76.9%
	Negative	50.35%	52%	23.1%
SVM	Positive	50.6%	50.92%	78.7%
	Negative	49.4%	49.07%	21.3%
SO-PMI-IR	Positive	52.37%	50.78%	59.1%
	Negative	47.63%	49.22%	40.9%
SWN (VS)	Positive	65.15%	64.85%	71.36%
	Negative	34.85%	35.07%	28.64%
SWN (APS)	Positive	65.9%	65.21%	70.05%
	Negative	34.1%	34.71%	29.95%
SWN (VS+APS)	Positive	63.8%	64.85%	69.57%
	Negative	32.95%	33.28%	29.47%

TABLE V. RESULT OF THREE DIFFERENT VARIANTS OF SENTIWORDNET IMPLEMENTATIONS ON DATASET1 AND DATASET2

Method	*Performance measure*		*Number*
SWN (VS)	1000 Positive Reviews	Classified Positive	725
		Classified Negative	275
	1000 Negative Reviews	Classified Negative	578
		Classified Positive	422
	700 Positive Reviews	Classified Positive	498
		Classified Negative	201
	700 Negative Reviews	Classified Negative	410
		Classified Positive	290
SWN (APS)	1000 Positive Reviews	Classified Positive	738
		Classified Negative	262
	1000 Negative Reviews	Classified Negative	580
		Classified Positive	420
	700 Positive Reviews	Classified Positive	507
		Classified Negative	192
	700 Negative Reviews	Classified Negative	406
		Classified Positive	294
SWN (VS+APS)	1000 Positive Reviews	Classified Positive	717
		Classified Negative	254
	1000 Negative Reviews	Classified Negative	563
		Classified Positive	405
	700 Positive Reviews	Classified Positive	492
		Classified Negative	186
	700 Negative Reviews	Classified Negative	396
		Classified Positive	280

Authorized licensed use limited to: b-on: UNIVERSIDADE DE AVEIRO. Downloaded on March 19,2025 at 19:55:39 UTC from IEEE Xplore. Restrictions apply.

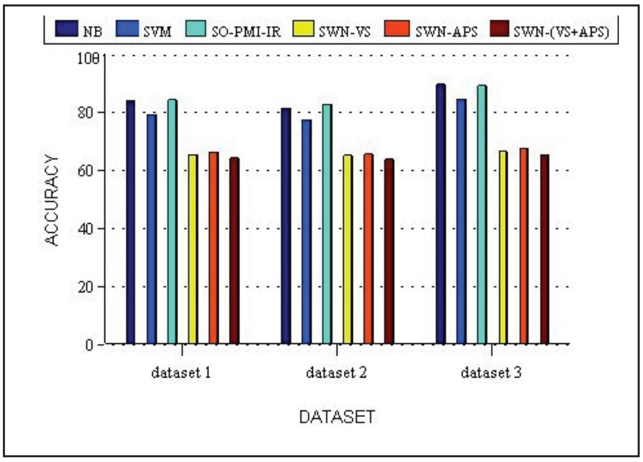

FIGURE 1: Plot of Accuracy values for the six versions implemented.

VII. OBSERVATIONS AND CONCLUSION

The analytical results obtained present a detailed evaluative account of performance of the different supervised machine learning based classifiers, unsupervised semantic orientation approach and the SentiWordnet approaches for sentiment analysis of movie reviews. As we can see from tables I, II and III, the accuracy of classification by NB is marginally better than the SVM and is close to the SO-PMI-IR algorithm. For the third dataset the SVM performance levels are identical to the NB and SO-PMI-IR. SentiWordNet on the other hand achieves a lower accuracy score.One thing that we can safely and definitely conclude is that NB performance can be comparable to the popularly believed superior performance of SVM, at least for sentiment classification. The SO-PMI-IR algorithm has obtained impressive accuracy levels (after agreeing on to the most suitable aggregation scheme) and seems the best choice due to its unsupervised nature, but the disadvantage is that we need to compute lot of PMI values, which itself is a time consuming and involved task. The SentiWordNet is computationally most favorable algorithm but achieves relatively lower accuracy levels and in simplest amount of computational time and without the need for any training.

Although there has been some previous work on evaluative account of some of these techniques [6], [20], including a previous work of our own [21]; this experimental work is different and unique in its design, setup and approach. It presents a comprehensive account of use of all the three available approaches for sentiment classification in the domain of movie reviews. The ease of implementation of SentiWordNet allows us to use it as an added level of filtering for movie recommendations. We can analyze all the reviews of a movie and if the majority reviews are classified as ‘positive’, it may be rated as ‘worth watch’ and consequently recommended to users. Though different approaches have their own promises, advantages and applications, SentiWordNet has the advantage of being added to any system on the fly without the requirement of any training.

REFERENCES

[1] V. K. Singh, M. Mukherjee & G. K. Mehta, “Combining Collaborative Filtering and Sentiment Analysis for Improved Movie Recommendations”, In C. Sombattheera et. al. (Eds.): Multi-disciplinary Trends in Artificial Intelligence, LNAI 7080, Springer-Verlag, Berlin-Heidelberg, pp. 38-50, 2011.

[2] V. K. Singh, M. Mukherjee & G. K. Mehta, “Combining a Content Filtering Heuristic and Sentiment Analysis for Movie Recommendations”, In K.R. Venugopal & L.M. Patnaik (Eds.): ICIP 2011, Aug. 2011, CCIS 157, pp. 659-664, Springer, Heidelberg, 2011.

[3] K. Dave, S. Lawerence & D. Pennock,“ Mining the Peanut Gallery-Opinion Extraction and Semantic Classification of Product Reviews”, Proceedings of the 12th International World Wide Web Conference, pp. 519-528, 2003.

[4] P. Turney, “Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews”, Proceedings of ACL-02, 40th Annual Meeting of the Association for Computational Linguistics, pp. 417-424, Philadelphia, US, 2002.

[5] A. Esuli & F.Sebastiani, “Determining the Semantic Orientation of terms through gloss analysis”, Proceedings of CIKM-05, 14th ACM International Conference on Information and Knowledge Management, pp. 617-624, Bremen, DE, 2005.

[6] B. Pang, L. Lee & S. Vaithyanathan, “Thumbs up? Sentiment classificationusing machine learning techniques”, Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 79-86, Philadelphia, US, 2002.

[7] S.M. Kim & E. Hovy, “Determining sentiment of opinions”, Proceedings of the COLING Conference, Geneva, 2004.

[8] K.T. Durant& M.D. Smith, “Mining Sentiment Classification from Political Web Logs”, Proceedings of WEBKDD’06, ACM, 2006.

[9] F. Sebastiani, “Machine Learning in Automated text categorization”, ACM Computing Surveys, 34(1): 1-47, 2002.

[10] P. Turney& M.L. Littman, “Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word corpus”, NRC Publications Archive, 2002.

[11] C.D. Manning, P. Raghavan & H. Schutze, “Introduction to Information Retrieval”, Cambridge University Press, New York, USA, 2008.

[12] V. K. Singh, M. Mukherjee, G. K. Mehta, N. Tiwari&S. Garg, “Opinion Mining from Weblogs and its Relevance for Socio-political Research”, In M. Natarajanet. al. (Eds.) Advances in Computer Science and Information Technology. Part II, LNICST 85, Springer, pp. 134-145, 2012.

[13] Weka Data Mining Software in JAVA, http://www.cs.waikato.ac.nz/ml/weka/

[14] B. Liu, “Web Data Mining: Exploring Hyperlinks, Contents and Usage Data”, Springer-Verlag, Berlin-Heidelberg, pp. 411-416, 2002.

[15] V. K. Singh, M. Mukherjee & G. K. Mehta, “Sentiment and Mood Analysis of Weblogs using POS Tagging based Approach”, In S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 313-324, Springer-Verlag, Berlin Heidelberg, 2011.

[16] SentiWordNet, available at http://www.sentiwordnet.isti.cnr.it

[17] F. Benamara, C. Cesarano& D. Reforigiato, “Sentiment Analysis: Adjectives and Adverbs are better than Adjectives Alone”, Proceedings of ICWSM 2006, CO USA, 2006.

[18] http://www.cs.cornell.edu/people/pabo/movie-review-data/

[19] Internet Movie Database, http://www.imdb.com

[20] P. Chaovalit & L. Zhou, “Movie Review Mining: Comparison between Supervised and Unsupervised Approaches”, Proceedings of the 38th Hawaii International Conference on System Sciences, 2005.

[21] P. Waila, Marisha, V.K. Singh & M.K. Singh, “Evaluating Machine Learning and Unsupervised Semantic Orientation Approaches for Sentiment Analysis of Textual Reviews”, Proceedings of International Conference on Computational Intelligence and Computing Research, Coimbatore, India, 2012.

Authorized licensed use limited to: b-on: UNIVERSIDADE DE AVEIRO. Downloaded on March 19,2025 at 19:55:39 UTC from IEEE Xplore. Restrictions apply.