Harnessing WordNet Senses for Supervised Sentiment Classification

Balamurali A R[1,2] **Aditya Joshi**[2] **Pushpak Bhattacharyya**[2]
[1] IITB-Monash Research Academy, IIT Bombay
[2]Dept. of Computer Science and Engineering, IIT Bombay
Mumbai, India - 400076
{balamurali,adityaj,pb}@cse.iitb.ac.in

Abstract

Traditional approaches to sentiment classification rely on lexical features, syntax-based features or a combination of the two. We propose semantic features using word senses for a supervised document-level sentiment classifier. To highlight the benefit of sense-based features, we compare word-based representation of documents with a sense-based representation where WordNet senses of the words are used as features. In addition, we highlight the benefit of senses by presenting a part-of-speech-wise effect on sentiment classification. Finally, we show that even if a WSD engine disambiguates between a limited set of words in a document, a sentiment classifier still performs better than what it does in absence of sense annotation. Since word senses used as features show promise, we also examine the possibility of using similarity metrics defined on WordNet to address the problem of not finding a sense in the training corpus. We perform experiments using three popular similarity metrics to mitigate the effect of unknown synsets in a test corpus by replacing them with *similar* synsets from the training corpus. The results show promising improvement with respect to the baseline.

1 Introduction

Sentiment Analysis (SA) is the task of prediction of opinion in text. Sentiment classification deals with tagging text as positive, negative or neutral from the perspective of the speaker/writer with respect to a topic. In this work, we follow the definition of Pang et al. (2002) & Turney (2002) and consider a binary classification task for output labels as positive and negative.

Traditional supervised approaches for SA have explored lexeme and syntax-level units as features. Approaches using lexeme-based features use bag-of-words (Pang and Lee, 2008) or identify the roles of different *parts-of-speech (POS)* like adjectives (Pang et al., 2002; Whitelaw et al., 2005). Approaches using syntax-based features construct parse trees (Matsumoto et al., 2005) or use text parsers to model valence shifters (Kennedy and Inkpen, 2006).

Our work explores incorporation of semantics in a supervised sentiment classifier. We use the synsets in Wordnet as the feature space to represent word senses. Thus, a document consisting of words gets mapped to a document consisting of corresponding word senses. Harnessing WordNet senses as features helps us address two issues:

1. Impact of WordNet sense-based features on the performance of supervised SA
2. Use of WordNet similarity metrics to solve the problem of features unseen in the training corpus

The first points deals with evaluating sense-based features against word-based features. The second issue that we address is in fact an opportunity to improve the performance of SA that opens up because of the choice of sense space. Since sense-based features prove to generate superior sentiment classifiers, we get an opportunity to mitigate unknown

synsets in the test corpus by replacing them with known synsets in the training corpus. Note that such replacement is not possible if word-based representation were used as it is computationaly not possible to make such large number of similarity comparisons.

We use the corpus by Ye et al. (2009) that consists of travel domain reviews marked as positive or negative at the document level. Our experiments on studying the impact of Wordnet sense-based features deal with variants of this corpus manually or automatically annotated with senses. Besides showing the overall impact, we perform a POS-wise analysis of the benefit to SA. In addition, we compare the effect of varying training samples on a sentiment classifier developed using word based features and sense based features. Through empirical evidence, we also show that disambiguating some words in a document also provides a better accuracy as compared to not disambiguating any words. These four sets of experiments highlight our hypothesis that WordNet senses are better features as compared to words.

Wordnet sense-based space allows us to mitigate unknown features in the test corpus. Our synset replacement algorithm uses Wordnet similarity-based metrics which replace an unknown synset in the test corpus with the closest approximation in the training corpus. Our results show that such a replacement benefits the performance of SA.

The roadmap for the rest of the paper is as follows: Existing related work in SA and the differentiating aspects of our work are explained in section 2 Section 3 describes the sense-based features that we use for this work. We explain the similarity-based replacement technique using WordNet synsets in section 4. Our experiments have been described in section 5. In section 6, we present our results and related discussions. Section 7 analyzes some of the causes for erroneous classification. Finally, section 8 concludes the paper and points to future work.

2 Related Work

This work studies the benefit of a word sense-based feature space to supervised sentiment classification. However, a word sense-based feature space is feasible subject to verification of the hypothesis that sentiment and word senses are related. Towards this, Wiebe and Mihalcea (2006) conduct a study on human annotation of 354 words senses with polarity and report a high inter-annotator agreement. The work in sentiment analysis using sense-based features, including ours, assumes this hypothesis that *sense decides the sentiment*.

The novelty of our work lies in the following. Firstly our approach is distinctly. Akkaya et al. (2009) and Martn-Wanton et al. (2010) report performance of rule-based sentiment classification using word senses. Instead of a rule-based implementation, We used supervised learning. The supervised nature of our approach renders lexical resources unnecessary as used in Martn-Wanton et al. (2010). Rentoumi et al. (2009) suggest using word senses to detect sentence level polarity of news headlines. The authors use graph similarity to detect polarity of senses. To predict sentence level polarity, a HMM is trained on word sense and POS as the observation. The authors report that word senses particularly help understanding metaphors in these sentences. Our work differs in terms of the corpus and document sizes in addition to generating a general purpose classifier.

Another supervised approach of creating an emotional intensity classifier using concepts as features has been reported by Carrillo de Albornoz et al. (2010). This work is different based on the feature space used. The concepts used for the purpose are limited to affective classes. This restricts the size of the feature space to a limited set of labels. As opposed to this, we construct feature vectors that map to a larger sense-based space. In order to do so, we use synset offsets as representation of sense-based features.

Akkaya et al. (2009), Martn-Wanton et al. (2010) and Carrillo de Albornoz et al. (2010) perform sentiment classification of individual sentences. However, we consider a document as a unit of sentiment classification *i.e.* our goal is to predict a document on the whole as positive or negative. This is different from Pang and Lee (2004) which suggests that sentiment is associated only with subjective content. A document in its entirety is represented using sense-based features in our experiments. Carrillo de Albornoz et al. (2010) suggests expansion using WordNet relations which we also follow. This is a benefit that can be achieved only in a sense-based space.

3 Features based on WordNet Senses

In their original form, documents are said to be in lexical space since they consist of words. When the words are replaced by their corresponding senses, the resultant document is said to be in semantic space.

WordNet 2.1 (Fellbaum, 1998) has been used as the sense repository. Each word/lexeme is mapped to an appropriate synset in WordNet based on its sense and represented using the corresponding synset id of WordNet. Thus, the word *love* is disambiguated and replaced by the identifier *21758160* which consists of a POS category identifier *2* followed by synset offset identifier *1758160*. This paper refers to synset offset as synset identifiers or simply, senses.

This section first gives the motivation for using word senses and then, describes the approaches that we use for our experiments.

3.1 Motivation

Consider the following sentences as the first scenario.

1. *"Her face **fell** when she heard that she had been fired."*

2. *"The fruit **fell** from the tree."*

The word '*fell*' occurs in different senses in the two sentences. In the first sentence, '*fell*' has the meaning of '*assume a disappointed or sad expression*, whereas in the second sentence, it has the meaning of '*descend in free fall under the influence of gravity*'. A user will infer the negative polarity of the first sentence from the negative sense of '*fell*' in it while the user will state that the second sentence does not carry any sentiment. This implies that there is at least one sense of the word '*fell*' that carries sentiment and at least one that does not.

In the second scenario, consider the following examples.

1. *"The snake bite proved to be **deadly** for the young boy."*

2. *"Shane Warne is a **deadly** spinner."*

The word *deadly* has senses which carry opposite polarity in the two sentences and these senses assign the polarity to the corresponding sentence. The first sentence is negative while the second sentence is positive.

Finally in the third scenario, consider the following pair of sentences.

1. *"He speaks a **vulgar** language."*

2. *"Now that's real **crude** behavior!"*

The words *vulgar* and *crude* occur as synonyms in the synset that corresponds to the sense '*conspicuously and tastelessly indecent*'. The synonymous nature of words can be identified only if they are looked at as senses and not just words.

As one may observe, the first scenario shows that a word may have some sentiment-bearing and some non-sentiment-bearing senses. In the second scenario, we show that there may be different senses of a word that bear sentiments of opposite polarity. Finally, in the third scenario, we show how a sense can be manifested using different words, *i.e.*, words in a synset. The three scenarios motivate the use of semantic space for sentiment prediction.

3.2 Sense versus Lexeme-based Feature Representation

We annotate the words in the corpus with their senses using two sense disambiguation approaches.

As the first approach, **manual sense annotation** of documents is carried out by two annotators on two subsets of the corpus, the details of which are given in Section 5.1. This is done to determine the ideal case scenario- the skyline performance.

As the second approach, a state-of-art algorithm for domain-specific WSD proposed by Khapra et al. (2010) is used to obtain an automatically sense-tagged corpus. This algorithm called **iterative WSD or IWSD** iteratively disambiguates words by ranking the candidate senses based on a scoring function.

The two types of sense-annotated corpus lead us to four feature representations for a document:

1. Word senses that have been manually annotated (*M*)

2. Word senses that have been annotated by an automatic WSD (*I*)

3. *Manually* annotated word senses and words (both separately as features) (*Words + Sense(M)*)

4. *Automatically* annotated word senses and words (both separately as features) (*Words + Sense(I)*)

Our first set of experiments compares the four feature representations to find the feature representation with which sentiment classification gives the best performance. W+S(M) and W+S(I) are used to overcome non-coverage of WordNet for some noun synsets. In addition to this, we also present a part-of-speech-wise analysis of benefit to SA as well as effect of varying the training samples on sentiment classification accuracy.

3.3 Partial disambiguation as opposed to no disambiguation

The state-of-the-art automatic WSD engine that we use performs (approximately) with 70% accuracy on tourism domain (Khapra et al., 2010). This means that the performance of SA depends on the performance of WSD which is not very high in case of the engine we use.

A partially disambiguated document is a document which does not contain senses of all words. Our hypothesis is that disambiguation of even few words in a document can give better results than no disambiguation. To verify this, we create different variants of the corpus by disambiguating words which have candidate senses within a threshold. For example, a partially disambiguated variant of the corpus with threshold 3 for candidate senses is created by disambiguating words which have *a maximum of three candidate senses*. These synsets are then used as features for classification along with lexeme based features. We conduct multiple experiments using this approach by varying the number of candidate senses.

4 Advantage of senses: Similarity Metrics and Unknown Synsets

4.1 Synset Replacement Algorithm

Using WordNet senses provides an opportunity to use similarity-based metrics for WordNet to reduce the effect of unknown features. If a synset encountered in a test document is not found in the training corpus, it is replaced by one of the synsets present in the training corpus. The substitute synset is determined on the basis of its similarity with the synset in the test document. The synset that is replaced is referred to as an *unseen synset* as it is not known to the trained model.

For example, consider excerpts of two reviews, the first of which occurs in the training corpus while the second occurs in the test corpus.

1. " *In the night, it is a* ***lovely*** *city and...* "

2. " *The city has many* ***beautiful*** *hot spots for honeymooners.* "

The synset of '*beautiful*' is not present in the training corpus. We evaluate a similarity metric for all synsets in the training corpus with respect to the sense of *beautiful* and find that the sense of *lovely* is closest to it. Hence, the sense of *beautiful* in the test document is replaced by the sense of *lovely* which is present in the training corpus.

The replacement algorithm is described in Algorithm 1. The algorithm follows from the fact that the similarity value for a synset with itself is maximum.

4.2 Similarity metrics used

We conduct different runs of the replacement algorithm using three similarity metrics, namely LIN's similarity metric, Lesk similarity metric and Leacock and Chodorow (LCH) similarity metric. These runs generate three variants of the corpus. We compare the benefit of each of these metrics by studying their sentiment classification performance. The metrics can be described as follows:

LIN: The metric by Lin (1998) uses the information content individually possessed by two concepts in addition to that shared by them. The information content shared by two concepts A and B is given by their most specific subsumer (lowest superordinate(*lso*). Thus, this metric defines the similarity between two concepts as

$$sim_{LIN}(A,B) = \frac{2 \times \log Pr(lso(A,B))}{\log Pr(A) + \log Pr(B)} \quad (1)$$

```
Input: Training Corpus, Test Corpus,
Similarity Metric
Output: New Test Corpus
T:= Training Corpus;
X:= Test Corpus;
S:= Similarity metric;
train_concept_list = get_list_concept(T) ;
test_concept_list = get_list_concept(X);
for each concept C in test_concept_list do
    temp_max_similarity = 0 ;
    temp_concept = C ;
    for each concept D in train_concept_list do
        similarity_value = get_similarity_value(C,D,S);
        if (similarity_value > temp_max_similarity) then
            temp_max_similarity= similarity_value;
            temp_concept = D ;
        end
    end
    C = temp_concept ;
    replace_synset_corpus(C,X);
end
Return X ;
```

Algorithm 1: Synset replacement using similarity metric

Lesk: Each concept in WordNet is defined through gloss. To compute the Lesk similarity (Banerjee and Pedersen, 2002) between A and B, a scoring function based on the overlap of words in their individual glosses is used.

Leacock and Chodorow (LCH): To measure similarity between two concepts A and B, Leacock and Chodorow (1998) compute the shortest path through hypernymy relation between them under the constraint that there exists such a path. The final value is computed by scaling the path length by the overall taxonomy depth (D).

$$sim_{LCH}(A,B) = -\log\left(\frac{len(A,B)}{2D}\right) \quad (2)$$

5 Experimentation

We describe the variants of the corpus generated and the experiments in this section.

5.1 Data Preparation

We create different variants of the dataset by Ye et al. (2009). This dataset contains 600 positive and 591 negative reviews about seven travel destinations. Each review contains approximately 4-5 sentences with an average number of words per review being 80-85.

To create the manually annotated corpus, two human annotators annotate words in the corpus with senses for two disjoint subsets of the original corpus by Ye et al. (2009). The inter-annotation agreement for a subset of the corpus showed 91% sense overlap. The manually annotated corpus consists of 34508 words with 6004 synsets.

POS	#Words	P(%)	R(%)	F-Score(%)
Noun	12693	75.54	75.12	75.33
Adverb	4114	71.16	70.90	71.03
Adjective	6194	67.26	66.31	66.78
Verb	11507	68.28	67.97	68.12
Overall	34508	71.12	70.65	70.88

Table 1: Annotation Statistics for IWSD; P- Precision,R-Recall

The second variant of the corpus contains word senses obtained from automatic disambiguation using IWSD. The evaluation statistics of the IWSD is shown in Table 1. Table 1 shows that the F-score for noun synsets is high while that for adjective synsets is the lowest among all. The low recall for adjective POS based synsets can be detrimental to classification since adjectives are known to express direct sentiment (Pang et al., 2002). Hence, in the context of sentiment classification, disambiguation of adjective synsets is more critical as compared to disambiguation of noun synsets.

5.2 Experimental setup

The experiments are performed using C-SVM (linear kernel with default parameters[1]) available as a part of LibSVM[2] package. We choose to use SVM since it performs the best for sentiment classification (Pang et al., 2002). All results reported are average of five-fold cross-validation accuracies.

To conduct experiments on words as features, we first perform stop-word removal. The words are not stemmed since stemming is known to be detrimental to sentiment classification (Leopold and Kindermann, 2002). To conduct the experiments based on

[1] C=0.0,ϵ=0.0010

[2] http://www.csie.ntu.edu.tw/ cjlin/libsvm

Feature Representations	Accuracy(%)	PF	NF	PP	NP	PR	NR
Words (Baseline)	84.90	85.07	84.76	84.95	84.92	85.19	84.60
Sense (M)	89.10	88.22	89.11	91.50	87.07	85.18	91.24
Words + Sense (M)	90.20	89.81	90.43	92.02	88.55	87.71	92.39
Sense (I)	85.48	85.31	85.65	87.17	83.93	83.53	87.46
Words + Sense (I)	86.08	86.28	85.92	85.87	86.38	86.69	85.46

Table 2: Classification Results; PF-Positive F-score(%), NF-Negative F-score (%), PP-Positive Precision (%), NP-Negative Precision (%), PR-Positive Recall (%), NR-Negative Recall (%)

the synset representation, words in the corpus are annotated with synset identifiers along with POS category identifiers. For automatic sense disambiguation, we used the trained IWSD engine from Khapra et al. (2010). These synset identifiers along with POS category identifiers are then used as features. For replacement using semantic similarity measures, we used WordNet::Similarity 2.05 package by Pedersen et al. (2004).

To evaluate the result, we use accuracy, F-score, recall and precision as the metrics. Classification accuracy defines the ratio of the number of true instances to the total number of instances. Recall is calculated as a ratio of the true instances found to the total number of false positives and true positives. Precision is defined as the number of true instances divided by number of true positives and false negatives. Positive Precision (PP) and Positive Recall (PR) are precision and recall for positive documents while Negative Precision (NP) and Negative Recall (NR) are precision and recall for negative documents. F-score is the weighted precision-recall score.

6 Results and Discussions

6.1 *Comparison of various feature representations*

Table 2 shows results of classification for different feature representations. The baseline for our results is the unigram bag-of-words model (Baseline).

An improvement of 4.2% is observed in the accuracy of sentiment prediction when manually annotated sense-based features (M) are used in place of word-based features (Words). The precision of both the classes using features based on semantic space is also better than one based on lexeme space. While reported results suggest that it is more difficult to detect negative sentiment than positive sentiment (Gindl and Liegl, 2008), our results show that negative recall increases by around 8% in case of sense-based representation of documents.

The combined model of words and manually annotated senses (Words + Senses (M)) gives the best performance with an accuracy of 90.2%. This leads to an improvement of 5.3% over the baseline accuracy [3].

One of the reasons for improved performance is the feature abstraction achieved due to the synset-based features. The dimension of feature vector is reduced by a factor of 82% when the document is represented in synset space. The reduction in dimensionality may also lead to reduction in noise (Cunningham, 2008).

A comparison of accuracy of different sense representations in Table 2 shows that manual disambiguation performs better than using automatic algorithms like IWSD. Although overall classification accuracy improvement of IWSD over baseline is marginal, negative recall also improves. This benefit is despite the fact that evaluation of IWSD engine over manually annotated corpus gave an overall F-score of 71% (refer Table 1). For a WSD engine with a better accuracy, the performance of sense-based SA can be boosted further.

Thus, in terms of feature representation of documents, sense-based features provide a better overall performance as compared to word-based features.

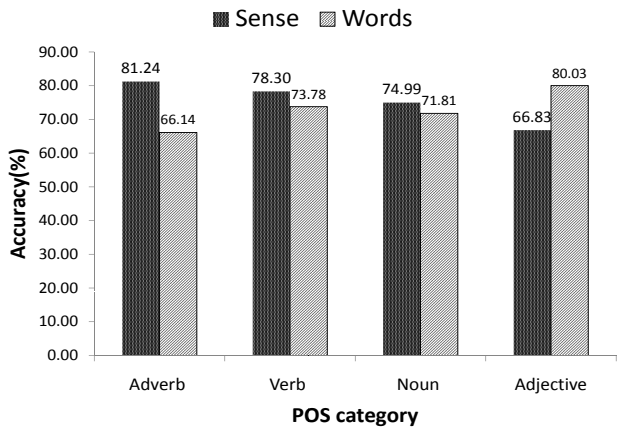

Figure 1: POS-wise statistics of manually annotated semantic space

6.2 *POS-wise analysis*

For each POS, we compare the performance of two models:

- Model trained on words of only that POS
- Model trained on word senses of only that POS

Figure 1 shows the parts-of-speech-wise classification accuracy of sentiment classification for senses (manual) and words. In the lexeme space, adjectives directly impact the classification performance. But it can be seen that disambiguation of adverb and verb synsets impact the performance of SA higher than disambiguation of nouns and adjectives.

While it is believed that adjectives carry direct sentiments, our results suggest that using adjectives alone as features may not improve the accuracy. The results prove that sentiment may be subtle at times and not expressed directly through adjectives.

As manual sense annotation is an effort and cost intensive process, the parts-of-speech-wise results suggest improvements expected from an automatic WSD engine so that it can aid sentiment classification. Table 1 suggests that the WSD engine works better for noun synsets compared to adjective and adverb synsets. While this is expected in a typical WSD setup, it is the adverbs and verbs that are more important for detecting sentiment in semantics space than nouns. The future WSD systems will have to show an improvement in their accuracy with respect to adverb and verb synsets.

[3]The improvement in results of semantic space is found to be statistically significant over the baseline at 95% confidence level when tested using a paired t-test.

	Sense		Words	
POS Category	**PF**	**NF**	**PF**	**NF**
Adverb	79.65	80.45	70.25	73.68
Verb	75.50	79.28	62.23	63.12
Noun	73.39	75.40	69.77	72.55
Adjective	63.11	65.03	78.29	79.20

Table 3: POS-wise F-score for sense (M) and Words;PF-Positive F-score(%), NF- Negative F-score (%)

Table 3 shows the positive and negative F-score statistics with respect to different POS. Detection of negative reviews using lexeme space is difficult. POS-wise statistics also suggest the same. It should be noted that adverb and verb synsets play an important role in negative class detection. Thus, an automatic WSD engine should give importance to the correct disambiguation of these POS categories.

6.3 *Effect of size of training corpus*

#Training Documents	**W**	**M**	**I**	**W+S(M)**	**W+S(I)**
100	76.5	87	79.5	82.5	79.5
200	81.5	88.5	82	90	84
300	79.5	92	81	89.5	82
400	82	90.5	81	94	85.5
500	83.5	91	85	96	82.5

Table 4: Accuracy (%) with respect to number of training documents; W: Words, M: Manual Annotation, I: IWSD-based sense annotation, W+S(M): Word+Senses (Manual annotation), W+S(I): Word+Senses(IWSD-based sense annotation)

From table 2, the benefit of sense disambiguation to sentiment prediction is evident. In addition, Table 4 shows variation of classification accuracy with respect to different number of training samples based on different approaches of annotation explained in previous sections. The results are based on a blind set of 90 test samples from both the polarity labels [4].

[4]No cross validation is performed for this experiment

Compared to lexeme-based features, manually annotated sense based features give better performance with lower number of training samples. IWSD is also better than lexeme-based features. A SA system trained on 100 training samples using manually annotated senses gives an accuracy of 87%. Word-based features never achieve this accuracy. An IWSD-based system requires lesser samples when compared to lexeme space for an equivalent accuracy. Note that model based on words + senses(M) features achieve an accuracy of 96% on this test set.

This implies that the synset space, in addition to benefit to sentiment prediction in general, requires *lesser number of training samples* in order to achieve the accuracy that lexeme space can achieve with a larger number of samples.

6.4 *Effect of Partial disambiguation*

Figure 2 shows the accuracy, positive F-score and negative F-score with respect to different thresholds of candidate senses for partially disambiguated documents as described in Section 3.3. We compare the performance of these documents with word-based features (B) and sense-based features based on manually (M) or automatically obtained senses (I). Note that Sense (I) and Sense (M) correspond to completely disambiguated documents.

In case of partial disambiguation using manual annotation, disambiguating words with less than three candidate senses performs better than others. For partial disambiguation that relies on an automatic WSD engine, a comparable performance to full disambiguation can be obtained by disambiguating words which have a maximum of four candidate senses.

As expected, completely disambiguated documents provide the best F-score and accuracy figures[5]. However, a performance comparable to complete disambiguation can be attained by disambiguating selective words.

Our results show that even if highly ambiguous (in terms of senses) words are not disambiguated by a WSD engine, the performance of sentiment classification improves.

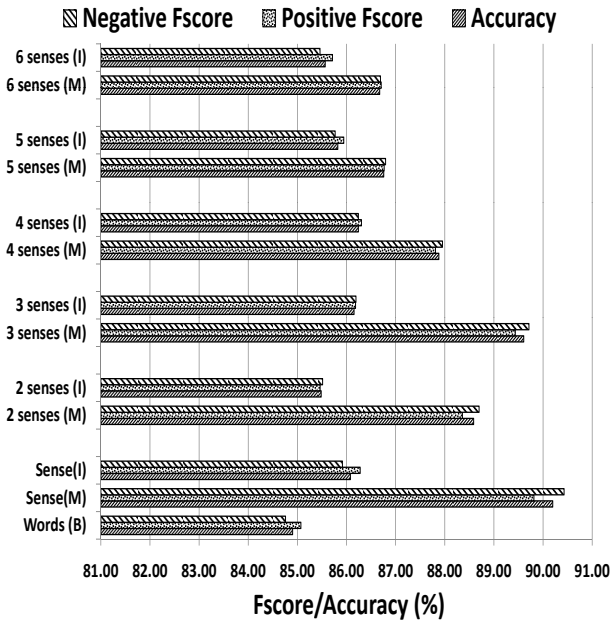

Figure 2: Partial disambiguation statistics: Accuracy,Positive F-score, Negative F-score variation with respect to sense disambiguation difficult level is shown. Words(B): baseline system

6.5 *Synset replacement using similarity metrics*

Table 5 shows the results of synset replacement experiments performed using similarity metrics defined in section 4. The similarity metric value NA shown in the table indicates that synset replacement is not performed for the specific run of experiment. For this set of experiments, we use the combination of sense and words as features (indicated by *Senses+Words (M)*).

Synset replacement using a similarity metric shows an improvement over using words alone. However, the improvement in classification accuracy is marginal compared to sense-based representation without synset replacement (Similarity Metric=NA).

Replacement using LIN and LCH metrics gives marginally better results compared to the vanilla setting in a manually annotated corpus. The same phenomenon is seen in the case of IWSD based approach[6]. The limited improvement can be due to the fact that since LCH and LIN consider only IS-A

[5] All results are statistically significant with respect to baseline

[6] Results based on LCH and LIN similarity metric for automatic sense disambiguation is not statistically significant with α=0.05

Feature Representation	Similarity Metric	Accuracy	PF	NF	PP	NP	PR	NR
Words (Baseline)	**NA**	84.90	85.07	84.76	84.95	84.92	85.19	84.60
Words + Sense(M)	**NA**	90.20	89.81	90.43	92.02	88.55	87.71	92.39
Words + Sense(I)	**NA**	86.08	86.28	85.92	85.87	86.38	86.69	85.46
Words + Sense (M)	**LCH**	90.60	90.20	90.85	92.85	88.61	87.70	93.21
Words + Sense(M)	**LIN**	90.70	90.26	90.97	93.17	88.50	87.53	93.57
Words + Sense (M)	**Lesk**	91.12	90.70	91.38	93.55	88.97	88.03	93.92
Words + Sense (I)	**LCH**	85.66	85.85	85.52	85.67	85.76	86.02	85.28
Words + Sense(I)	**LIN**	86.16	86.37	86.00	86.06	86.40	86.69	85.61
Words + Sense (I)	**Lesk**	86.25	86.41	86.10	86.31	86.26	86.52	85.95

Table 5: Similarity Metric Analysis using different similarity metrics with synsets and a combinations of synset and words;PF-Positive F-score(%), NF-Negative F-score (%), PP-Positive Precision (%), NP-Negative Precision (%), PR-Positive Recall (%), NR-Negative Recall (%)

Top information content features (in %)	IWSD synset #	Manual synsets #	Match synset #	Match Synsets (%)	Unmatched Synset(%)
10	601	722	288	39.89	60.11
20	1199	1443	650	45.05	54.95
30	1795	2165	1005	46.42	53.58
40	2396	2889	1375	47.59	52.41
50	2997	3613	1730	47.88	52.12

Table 6: Comparison of top information gain-based features of manually annotated corpora and automatically annotated corpora

relationship in WordNet, the replacement happens only for verbs and nouns. This excludes adverb synsets which we have shown to be the best features for a sense-based SA system.

Among all similarity metrics, the best classification accuracy is achieved using Lesk. The system performs with an overall classification accuracy of 91.12%, which is a substantial improvement of 6.2% over baseline. Again, it is only 1% over the vanilla setting that uses combination of synset and words. However, the similarity metric is not sophisticated as LIN or LCH.

Thus, we observe a marginal improvement by using similarity-based metrics for WordNet. A good metric which covers all POS categories can provide substantial improvement in the classification accuracy.

7 Error Analysis

For sentiment classification based on semantic space, we classify the errors into four categories. The examples quoted are from manual evaluation of the results.

1. *Effect of low disambiguation accuracy of IWSD engine:* SA using automatic sense annotation depends on the annotation system used. To assess the impact of IWSD system on sentiment classification, we compare the feature set based on manually annotated senses with the feature set based on automatically annotated senses. We compare the most informative features of the two classifiers. Table 6 shows the number of top informative features (synset) selected as the percentage of total synset features present when the semantic representation of documentation is used. The matched synset column represents the number of IWSD synsets that match

with manually annotated synsets.

The number of top performing features is more in case of manually annotated synsets. This can be attributed to the total number of synsets tagged in the two variant of the corpus. The reduction in the performance of SA for automatically annotated senses is because of the number of unmatched synsets.

Thus, although the accuracy of IWSD is currently 70%, the table indicates that IWSD can match the performance of manually annotated senses for SA if IWSD is able to tag correctly those top information content synsets. This aspect needs to be investigated further.

2. *Negation Handling:* For the purpose of this work, we concentrate on words as units for sentiment determination. Syntax and its contribution in understanding sentiment is neglected and hence, positive documents which contain negations are wrongly classified as negative. Negation may be direct as in the excerpt '*....what is there not to like about Vegas.*' or may be double as in the excerpt'*...that aren't insecure'*.

3. *Interjections and WordNet coverage:* Recent informal words are not covered in WordNet and hence, do not get disambiguated. The same is the case for interjections like *'wow'*,*'duh'* which sometimes carry direct sentiment. Lexical resources which include them can be used to incorporate information about these lexical units.

4. *Document Specificity:* The assumption underlying our analysis is that a document contains description of only one topic. However, reviews are generic in nature and tend to express contrasting sentiment about sub-topics . For example, a travel review about Paris can talk about restaurants in Paris, traffic in Paris, public behaviour, etc. with opposing sentiments. Assigning an overall sentiment to a document is subjective in such cases.

8 Conclusion & Future Work

This work presents an empirical benefit of WSD to sentiment analysis. The study shows that supervised sentiment classifier modeled on wordNet senses perform better than word-based features. We show how the performance impact differs for different automatic and manual techniques, parts-of-speech, different training sample size and different levels of disambiguation. In addition, we also show the benefit of using WordNet based similarity metrics for replacing unknown features in the test set. Our results support the fact that not only does sense space improve the performance of a sentiment classification system, but also opens opportunities for improvement using better similarity metrics.

Incorporation of syntactical information along with semantics can be an interesting area of work. More sophisticated features which include the two need to be explored. Another line of work is in the context of cross-lingual sentiment analysis. Current solutions are based on machine translation which is very resource-intensive. Using a bi-lingual dictionary which maps WordNet across languages, such a machine translation sub-system can be avoided.

Acknowledgment

We thank Jaya Saraswati and Rajita Shukla from CFILT Lab, IIT Bombay for annotating the dataset used for this work. We also thank Mitesh Khapra and Salil Joshi, IIT Bombay for providing us with the IWSD engine for the required experiments.

References

Cem Akkaya, Janyce Wiebe, and Rada Mihalcea. 2009. Subjectivity word sense disambiguation. In *Proc. of EMNLP '09*, pages 190–199, Singapore.

Satanjeev Banerjee and Ted Pedersen. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Proc. of CICLing'02*, pages 136–145, London, UK.

Jorge Carrillo de Albornoz, Laura Plaza, and Pablo Gervs. 2010. Improving emotional intensity classification using word sense disambiguation. *Special issue: Natural Language Processing and its Applications. Journal on Research in Computing Science*, 46:131–142.

Pdraig Cunningham. 2008. Dimension reduction. In *Machine Learning Techniques for Multimedia*, Cognitive Technologies, pages 91–112.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Stefan Gindl and Johannes Liegl, 2008. *Evaluation of different sentiment detection methods for polarity classification on web-based reviews*, pages 35–43.

Alistair Kennedy and Diana Inkpen. 2006. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2):110–125.

Mitesh Khapra, Sapan Shah, Piyush Kedia, and Pushpak Bhattacharyya. 2010. Domain-specific word sense disambiguation combining corpus basedand wordnet based parameters. In *Proc. of GWC'10*, Mumbai, India.

Claudia Leacock and Martin Chodorow. 1998. Combining local context with wordnet similarity for word sense identification. In *WordNet: A Lexical Reference System and its Application*.

Edda Leopold and Jörg Kindermann. 2002. Text categorization with support vector machines. how to represent texts in input space? *Machine Learning*, 46:423–444.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *In Proc. of the 15th International Conference on Machine Learning*, pages 296–304.

Tamara Martn-Wanton, Alexandra Balahur-Dobrescu, Andres Montoyo-Guijarro, and Aurora Pons-Porrata. 2010. Word sense disambiguation in opinion mining: Pros and cons. In *Proc. of CICLing'10*, Madrid,Spain.

Shotaro Matsumoto, Hiroya Takamura, and Manabu Okumura. 2005. Sentiment classification using word sub-sequences and dependency sub-trees. In *Proc. of PAKDD'05,*, Lecture Notes in Computer Science, pages 301–311.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. of ACL'04*, pages 271–278, Barcelona, Spain.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2:1–135.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. volume 10, pages 79–86.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. Wordnet::similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL'04*, pages 38–41.

Vassiliki Rentoumi, George Giannakopoulos, Vangelis Karkaletsis, and George A. Vouros. 2009. Sentiment analysis of figurative language using a word sense disambiguation approach. In *Proc. of the International Conference RANLP'09*, pages 370–375, Borovets, Bulgaria.

Peter Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proc. of ACL'02*, pages 417–424, Philadelphia, US.

Casey Whitelaw, Navendu Garg, and Shlomo Argamon. 2005. Using appraisal groups for sentiment analysis. In *Proc. of CIKM '05*, pages 625–631, New York, NY, USA.

Janyce Wiebe and Rada Mihalcea. 2006. Word sense and subjectivity. In *Proc. of COLING-ACL'06*, pages 1065–1072.

Qiang Ye, Ziqiong Zhang, and Rob Law. 2009. Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Systems with Applications*, 36(3):6527 – 6535.