Sentiment Relevance

Christian Scheible
Institute for Natural Language Processing
University of Stuttgart, Germany
scheibcn@ims.uni-stuttgart.de

Hinrich Schütze
Center for Information
and Language Processing
University of Munich, Germany

Abstract

A number of different notions, including subjectivity, have been proposed for distinguishing parts of documents that convey sentiment from those that do not. We propose a new concept, *sentiment relevance*, to make this distinction and argue that it better reflects the requirements of sentiment analysis systems. We demonstrate experimentally that sentiment relevance and subjectivity are related, but different. Since no large amount of labeled training data for our new notion of sentiment relevance is available, we investigate two semi-supervised methods for creating sentiment relevance classifiers: a distant supervision approach that leverages structured information about the domain of the reviews; and transfer learning on feature representations based on lexical taxonomies that enables knowledge transfer. We show that both methods learn sentiment relevance classifiers that perform well.

1 Introduction

It is generally recognized in sentiment analysis that only a subset of the content of a document contributes to the sentiment it conveys. For this reason, some authors distinguish the categories *subjective* and *objective* (Wilson and Wiebe, 2003). Subjective statements refer to the internal state of mind of a person, which cannot be observed. In contrast, objective statements can be verified by observing and checking reality. Some sentiment analysis systems filter out objective language and predict sentiment based on subjective language only because objective statements do not directly reveal sentiment.

Even though the categories subjective/objective are well-established in philosophy, we argue that they are not optimal for sentiment analysis. We instead introduce the notion of *sentiment relevance* (*S-relevance* or *SR* for short). A sentence or linguistic expression is S-relevant if it contains information about the sentiment the document conveys; it is *S-nonrelevant* (*SNR*) otherwise.

Ideally, we would like to have at our disposal a large annotated training set for our new concept of sentiment relevance. However, such a resource does not yet exist. For this reason, we investigate two semi-supervised approaches to S-relevance classification that do not require S-relevance-labeled data. The first approach is distant supervision (DS). We create an initial labeling based on domain-specific metadata that we extract from a public database and show that this improves performance by 5.8% F_1 compared to a baseline. The second approach is transfer learning (TL) (Thrun, 1996). We show that TL improves F_1 by 12.6% for sentiment relevance classification when we use a feature representation based on lexical taxonomies that supports knowledge transfer.

In our approach, we classify *sentences* as S-(non)relevant because this is the most fine-grained level at which S-relevance manifests itself; at the word or phrase level, S-relevance classification is not possible because of scope and context effects. However, S-relevance is also a discourse phenomenon: authors tend to structure documents into S-relevant passages and S-nonrelevant passages. To impose this discourse constraint, we employ a sequence model. We represent each document as a graph of sentences and apply a minimum cut method.

The rest of the paper is structured as follows. Section 2 introduces the concept of sentiment relevance and relates it to subjectivity. In Section 3, we review previous work related to sentiment relevance. Next, we describe the methods applied in this paper (Section 4) and the features we extract (Section 5). Finally, we turn to the description and

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 954–963,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics

results of our experiments on distant supervision (Section 6) and transfer learning (Section 7). We end with a conclusion in Section 8.

2 Sentiment Relevance

Sentiment Relevance is a concept to distinguish content informative for determining the sentiment of a document from uninformative content. This is in contrast to the usual distinction between subjective and objective content. Although there is overlap between the two notions, they are different. Consider the following examples for subjective and objective sentences:

(1) Subjective example: *Bruce Banner, a genetics researcher with a tragic past, suffers a horrible accident.*

(2) Objective example: *The movie won a Golden Globe for best foreign film and an Oscar.*

Sentence (1) is subjective because assessments like *tragic past* and *horrible accident* are subjective to the reader and writer. Sentence (2) is objective since we can check the truth of the statement. However, even though sentence (1) has negative subjective content, it is not S-relevant because it is about the plot of the movie and can appear in a glowingly positive review. Conversely, sentence (2) contributes to the positive opinion expressed by the author. Subjectivity and S-relevance are two distinct concepts that do not imply each other: Generally neutral and objective sentences can be S-relevant while certain subjective content is S-nonrelevant. Below, we first describe the annotation procedure for the sentiment relevance corpus and then demonstrate empirically that subjectivity and S-relevance differ.

2.1 Sentiment Relevance Corpus

For our initial experiments, we focus on sentiment relevance classification in the movie domain. To create a sentiment-relevance-annotated corpus, the SR corpus, we randomly selected 125 documents from the movie review data set (Pang et al., 2002).[1] Two annotators annotated the sentences for S-relevance, using the labels `SR` and `SNR`. If no decision can be made because a sentence contains both S-relevant and S-nonrelevant linguistic material, it is marked as `uncertain`. We excluded 360 sentences that were labeled uncertain from the evaluation. In total, the SR corpus contains 2759 S-relevant and 728 S-nonrelevant sentences. Figure 1 shows an excerpt from the corpus. The full corpus is available online.[2]

First, we study agreement between human annotators. We had 762 sentences annotated for S-relevance by both annotators with an agreement (Fleiss' κ) of .69. In addition, we obtained subjectivity annotations for the same data on Amazon Mechanical Turk, obtaining each label through a vote of three, with an agreement of $\kappa = .61$. However, the agreement of the subjectivity and relevance labelings after voting, assuming that subjectivity equals relevance, is only at $\kappa = .48$. This suggests that there is indeed a measurable difference between subjectivity and relevance. An annotator who we asked to examine the 225 examples where the annotations disagree found that 83.5% of these cases are true differences.

2.2 Contrastive Classification Experiment

We will now examine the similarities of S-relevance and an existing subjectivity dataset. Pang and Lee (2004) introduced subjectivity data (henceforth *P&L corpus*) that consists of 5000 highly subjective (*quote*) review snippets from rottentomatoes.com and 5000 objective (*plot*) sentences from IMDb plot descriptions.

We now show that although the P&L selection criteria (quotes, plot) bear resemblance to the definition of S-relevance, the two concepts are different.

We use *quote* as S-relevant and *plot* as S-nonrelevant data in TL. We divide both the SR and P&L corpora into training (50%) and test sets (50%) and train a Maximum Entropy (MaxEnt) classifier (Manning and Klein, 2003) with bag-of-word features. Macro-averaged F_1 for the four possible training-test combinations is shown in Table 1. The results clearly show that the classes defined by the two labeled sets are different. A classifier trained on P&L performs worse by about 8% on SR than a classifier trained on SR (68.5 vs. 76.4). A classifier trained on SR performs worse by more than 20% on P&L than a classifier trained on P&L (67.4 vs. 89.7).

Note that the classes are not balanced in the S-relevance data while they are balanced in the subjectivity data. This can cause a misestimation

[1] We used the texts from the raw HTML files since the processed version does not have capitalization.

[2] `http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/sentimentrelevance/`

O	SNR	Braxton is a gambling addict in deep to Mook (Ellen Burstyn), a local bookie.
S	SNR	Kennesaw is bitter about his marriage to a socialite (Rosanna Arquette), believing his wife to be unfaithful.
S	SR	The plot is twisty and complex, with lots of lengthy flashbacks, and plenty of surprises.
S	SR	However, there are times when it is needlessly complex, and at least one instance the storytelling turns so muddled that the answers to important plot points actually get lost.
S	SR	Take a look at L. A. Confidential, or the film's more likely inspiration, The Usual Suspects for how a complex plot can properly be handled.

Figure 1: Example data from the SR corpus with subjectivity (S/O) and S-relevance (SR/SNR) annotations

		test	
		P&L	SR
train	P&L	89.7	68.5
	SR	67.4	76.4

Table 1: TL/in-task F_1 for P&L and SR corpora

vocabulary	fp_{SR}	fp_{SNR}
{actor, director, story}	0	7.5
{good, bad, great}	11.5	4.8

Table 2: % incorrect sentences containing specific words

of class probabilities and lead to the experienced performance drops. Indeed, if we either balance the S-relevance data or unbalance the subjectivity data, we can significantly increase F_1 to 74.8% and 77.9%, respectively, in the noisy label transfer setting. Note however that this step is difficult in practical applications if the actual label distribution is unknown. Also, in a real practical application the distribution of the data is what it is – it cannot be adjusted to the training set. We will show in Section 7 that using an unsupervised sequence model is superior to artificial manipulation of class-imbalances.

An error analysis for the classifier trained on P&L shows that many sentences misclassified as S-relevant (fp_{SR}) contain polar words; for example, *Then, the situation turns bad*. In contrast, sentences misclassified as S-nonrelevant (fp_{SNR}) contain named entities or plot and movie business vocabulary; for example, *Tim Roth delivers the most impressive acting job by getting the body language right*.

The word count statistics in Table 2 show this for three polar words and for three plot/movie business words. The P&L-trained classifier seems to have a strong bias to classify sentences with polar words as S-relevant even if they are not, perhaps because most training instances for the category *quote* are highly subjective, so that there is insufficient representation of less emphatic S-relevant sentences. These snippets rarely contain plot/movie-business words, so that the P&L-trained classifier assigns almost all sentences with such words to the category S-nonrelevant.

3 Related Work

Many publications have addressed subjectivity in sentiment analysis. Two important papers that are based on the original philosophical definition of the term (internal state of mind vs. external reality) are (Wilson and Wiebe, 2003) and (Riloff and Wiebe, 2003). As we argue above, if the goal is to identify parts of a document that are useful/non-useful for sentiment analysis, then S-relevance is a better notion to use.

Researchers have implicitly deviated from the philosophical definition because they were primarily interested in satisfying the needs of a particular task. For example, Pang and Lee (2004) use a minimum cut graph model for review summarization. Because they do not directly evaluate the results of subjectivity classification, it is not clear to what extent their method is able to identify subjectivity correctly.

In general, it is not possible to know what the underlying concepts of a statistical classification are if no detailed annotation guidelines exist and no direct evaluation of manually labeled data is performed.

Our work is most closely related to (Taboada et al., 2009) who define a fine-grained classification that is similar to sentiment relevance on the highest level. However, unlike our study, they fail to experimentally compare their classification scheme to prior work in their experiments and to show that this scheme is different. In addition, they work on the paragraph level. However, paragraphs often contain a mix of S-relevant and S-nonrelevant sentences. We use the minimum cut method and are therefore able to incorporate discourse-level constraints in a more flexible fashion, giving preference to “relevance-uniform” paragraphs without mandating them.

Täckström and McDonald (2011) develop a fine-grained annotation scheme that includes S-nonrelevance as one of five categories. However, they do not use the category S-nonrelevance directly in their experiments and do not evaluate classification accuracy for it. We do not use their data set as it would cause domain mismatch between the product reviews they use and the available movie review subjectivity data (Pang and Lee, 2004) in the TL approach. Changing both the domain (movies to products) and the task (subjectivity to S-relevance) would give rise to interactions that we would like to avoid in our study.

The notion of annotator rationales (Zaidan et al., 2007) has some overlap with our notion of sentiment relevance. Yessenalina et al. (2010) use rationales in a multi-level model to integrate sentence-level information into a document classifier. Neither paper presents a direct gold standard evaluation of the accuracy of rationale detection.

In summary, no direct evaluation of sentiment relevance has been performed previously. One contribution in this paper is that we provide a single-domain gold standard for sentiment relevance, created based on clear annotation guidelines, and use it for direct evaluation.

Sentiment relevance is also related to review mining (e.g., (Ding et al., 2008)) and sentiment retrieval techniques (e.g., (Eguchi and Lavrenko, 2006)) in that they aim to find phrases, sentences or snippets that are relevant for sentiment, either with respect to certain features or with a focus on high-precision retrieval (cf. (Liu, 2010)). However, finding a few S-relevant items with high precision is much easier than the task we address: exhaustive classification of all sentences.

Another contribution is that we show that generalization based on semantic classes improves S-relevance classification. While previous work has shown the utility of other types of feature generalization for sentiment and subjectivity analysis (e.g., syntax and part-of-speech (Riloff and Wiebe, 2003)), semantic classes have so far not been exploited.

Named-entity features in movie reviews were first used by Zhuang et al. (2006), in the form of feature-opinion pairs (e.g., a positive opinion about the acting). They show that recognizing plot elements (e.g., script) and classes of people (e.g., actor) benefits review summarization. We follow their approach by using IMDb to define named entity features. We extend their work by introducing methods for labeling partial uses of names and pronominal references. We address a different problem (S-relevance vs. opinions) and use different methods (graph-based and statistical vs. rule-based).

Täckström and McDonald (2011) also solve a similar sequence problem by applying a distantly supervised classifier with an unsupervised hidden sequence component. Their setup differs from ours as our focus lies on pattern-based distant supervision instead of distant supervision using documents for sentence classification.

Transfer learning has been applied previously in sentiment analysis (Tan and Cheng, 2009), targeting polarity detection.

4 Methods

Due to the sequential properties of S-relevance (cf. Taboada et al. (2009)), we impose the discourse constraint that an S-relevant (resp. S-nonrelevant) sentence tends to follow an S-relevant (resp. S-nonrelevant) sentence. Following Pang and Lee (2004), we use minimum cut (MinCut) to formalize this discourse constraint.

For a document with n sentences, we create a graph with $n + 2$ nodes: n sentence nodes and source and sink nodes. We define source and sink to represent the classes S-relevance and S-nonrelevance, respectively, and refer to them as SR and SNR.

The individual weight $\text{ind}(s, x)$ between a sentence s and the source/sink node $x \in \{\text{SR}, \text{SNR}\}$ is weighted according to some confidence measure for assigning it to the corresponding class. The weight on the edge from the document's i^{th} sentence s_i to its j^{th} sentence s_j is set to $\text{assoc}(s_i, s_j) = c/(j - i)^2$ where c is a parameter (cf. (Pang and Lee, 2004)). The minimum cut is a tradeoff between the confidence of the classification decisions and “discourse coherence”. The discourse constraint often has the effect that high-confidence labels are propagated over the sequence. As a result, outliers with low confidence are eliminated and we get a "smoother" label sequence.

To compute minimum cuts, we use the push-relabel maximum flow method (Cherkassky and Goldberg, 1995).[3]

We need to find values for multiple free parameters related to the sequence model. Supervised optimization is impossible as we do not have any labeled data. We therefore resort to a proxy measure, the *run count*. A run is a sequence of sentences with the same label. We set each parameter p to the value that produces a median run count that is closest to the true median run count (or, in case of a tie, closest to the true mean run count). We assume that the optimal median/mean run count is known. In practice, it can be estimated from a small number of documents. We find the optimal value of p by grid search.

5 Features

Choosing features is crucial in situations where no high-quality training data is available. We are interested in features that are robust and support generalization. We propose two linguistic feature types for S-relevance classification that meet these requirements.

5.1 Generalization through Semantic Features

Distant supervision and transfer learning are settings where exact training data is unavailable. We therefore introduce generalization features which are more likely to support knowledge transfer. To generalize over concepts, we use knowledge from taxonomies. A set of generalizations can be induced by making a cut in the taxonomy and defining the concepts there as base classes. For nouns, the taxonomy is WordNet (Miller, 1995) for which CoreLex (Buitelaar, 1998) gives a set of basic types. For verbs, VerbNet (Kipper et al., 2008) already contains base classes.

We add for each verb in VerbNet and for each noun in CoreLex its base class or basic type as an additional feature where words tagged by the mate tagger (Bohnet, 2010) as `NN.*` are treated as nouns and words tagged as `VB.*` as verbs. For example, the verb *suggest* occurs in the VerbNet base class *say*, so we add a feature `VN:say` to the feature representation. We refer to these feature sets as *CoreLex* (*CX*) and *VerbNet* (*VN*) features and to their combination as *semantic features* (*SEM*).

5.2 Named Entities

As standard named entity recognition (NER) systems do not capture categories that are relevant to the movie domain, we opt for a lexicon-based approach similar to (Zhuang et al., 2006). We use the IMDb movie metadata database[4] from which we extract names for the categories `<ACTOR>`, `<PERSONNEL>` (directors, screenwriters, and composers), and `<CHARACTER>` (movie characters). Many entries are unsuitable for NER, e.g., *dog* is frequently listed as a character. We filter out all words that also appear in lower case in a list of English words extracted from the dict.cc dictionary.[5]

A name n can be ambiguous between the categories (e.g., *John Williams*). We disambiguate by calculating the maximum likelihood estimate of $p(c|n) = \frac{f(n,c)}{\sum_{c'} f(n,c')}$ where c is one of the three categories and $f(n,c)$ is the number of times n occurs in the database as a member of category c. We also calculate these probabilities for all tokens that make up a name. While this can cause false positives, it can help in many cases where the name obviously belongs to a category (e.g., *Skywalker* in *Luke Skywalker* is very likely a character reference). We always interpret a name preceding an actor in parentheses as a character mention, e.g., *Reese Witherspoon* in *Tracy Flick (Reese Witherspoon) is an overachiever [. . .]* This way, we can recognize character mentions for which IMDb provides insufficient information.

In addition, we use a set of simple rules to propagate annotations to related terms. If a capitalized word occurs, we check whether it is part of an already recognized named entity. For example, if we encounter *Robin* and we previously encountered *Robin Hood*, we assume that the two entities match. Personal pronouns will match the most recently encountered named entity. This rule has precedence over NER, so if a name matches a labeled entity, we do not attempt to label it through NER.

The aforementioned features are encoded as binary presence indicators for each sentence. This

[3] using the HIPR tool (`www.avglab.com/andrew/soft.html`)

[4] `www.imdb.com/interfaces/`

[5] `dict.cc`

feature set is referred to as *named entities* (*NE*).

5.3 Sequential Features

Following previous sequence classification work with Maximum Entropy models (e.g., (Ratnaparkhi, 1996)), we use selected features of adjacent sentences. If a sentence contains a feature `F`, we add the feature `F+1` to the following sentence. For example, if a `<CHARACTER>` feature occurs in a sentence, `<CHARACTER+1>` is added to the following sentence. For S-relevance classification, we perform this operation only for NE features as they are restricted to a few classes and thus will not enlarge the feature space notably. We refer to this feature set as *sequential features* (*SQ*).

6 Distant Supervision

Since a large labeled resource for sentiment relevance classification is not yet available, we investigate semi-supervised methods for creating sentiment relevance classifiers. In this section, we show how to bootstrap a sentiment relevance classifier by distant supervision (DS) .

Even though we do not have sentiment relevance annotations, there are sources of metadata about the movie domain that we can leverage for distant supervision. Specifically, movie databases like IMDb contain both metadata about the plot, in particular the characters of a movie, and metadata about the "creators" who were involved in the production of the movie: actors, writers, directors, and composers. On the one hand, statements about characters usually describe the plot and are not sentiment relevant and on the other hand, statements about the creators tend to be evaluations of their contributions – positive or negative – to the movie. We formulate a classification rule based on this observation: Count occurrences of NE features and label sentences that contain a majority of creators (and tied cases) as SR and sentences that contain a majority of characters as SNR. This simple labeling rule covers 1583 sentences with an F_1 score of 67.2% on the SR corpus. We call these labels inferred from NE metadata *distant supervision (DS) labels*. This is a form of distant supervision in that we use the IMDb database as described in Section 5 to automatically label sentences based on which metadata from the database they contain.

To increase coverage, we train a Maximum Entropy (MaxEnt) classifier (Manning and Klein, 2003) on the labels. The MaxEnt model achieves an F_1 of 61.2% on the SR corpus (Table 3, line 2). As this classifier uses training data that is biased towards a specialized case (sentences containing the named entity types creators and characters), it does not generalize well to other S-relevance problems and thus yields lower performance on the full dataset. This distant supervision setup suffers from two issues. First, the classifier only sees a subset of examples that contain named entities, making generalization to other types of expressions difficult. Second, there is no way to control the quality of the input to the classifier, as we have no confidence measure for our distant supervision labeling rule. We will address these two issues by introducing an intermediate step, the unsupervised sequence model introduced in Section 4.

As described in Section 4, each document is represented as a graph of sentences and weights between sentences and source/sink nodes representing SR/SNR are set to the confidence values obtained from the distantly trained MaxEnt classifier. We then apply MinCut as described in the following paragraphs and select the most confident examples as training material for a new classifier.

6.1 MinCut Setup

We follow the general MinCut setup described in Section 4. As explained above, we assume that creators and directors indicate relevance and characters indicate nonrelevance. Accordingly, we define n_{SR} to be the number of `<ACTOR>` and `<PERSONNEL>` features occurring in a sentence, and n_{SNR} the number of `<CHARACTER>` features. We then set the individual weight between a sentence and the source/sink nodes to $\text{ind}(s, x) = n_x$ where $x \in \{\text{SR}, \text{SNR}\}$. The MinCut parameter c is set to 1; we wish to give the association scores high weights as there might be long spans that have individual weights with zero values.

6.2 Confidence-based Data Selection

We use the output of the base classifier to train supervised models. Since the MinCut model is based on a weak assumption, it will make many false decisions. To eliminate incorrect decisions, we only use documents as training data that were labeled with high confidence. As the confidence measure for a document, we use the maximum flow value f – the "amount of fluid" flowing through the document. The max-flow min-cut theorem (Ford and Fulkerson, 1956) implies that if the flow value

	Model	Features	F_{SR}	F_{SNR}	F_m
1	Majority BL	–	88.3	0.0	44.2
2	MaxEnt (DSlabels)	NE	79.8	42.6	61.2^{1}
3	DSlabels+MinCut	NE	79.6	48.2	63.9^{12}
4	DS MaxEnt	NE	84.8	46.4	65.6^{12}
5	DS MaxEnt	NE+SEM	85.2	48.0	66.6^{124}
6	DS CRF	NE	83.4	49.5	66.4^{12}
7	DS MaxEnt	NE+SQ	84.8	49.2	67.0^{1234}
8	DS MaxEnt	NE+SQ+SEM	84.5	49.1	66.8^{1234}

Table 3: Classification results: F_{SR} (S-relevant F_1), F_{SNR} (S-nonrelevant F_1), and F_m (macro-averaged F_1). Superscript numbers indicate a significant improvement over the corresponding line.

is low, then the cut was found more quickly and thus can be easier to calculate; this means that the sentence is more likely to have been assigned to the correct segment. Following this assumption, we train MaxEnt and Conditional Random Field (CRF, (McCallum, 2002)) classifiers on the $k\%$ of documents that have the lowest maximum flow values f, where k is a parameter which we optimize using the run count method introduced in Section 4.

6.3 Experiments and Results

Table 3 shows S-relevant (F_{SR}), S-nonrelevant (F_{SNR}) and macro average (F_m) F_1 values for different setups with this parameter. We compare the following setups: (1) The majority baseline (BL) i.e., choosing the most frequent label (SR). (2) a MaxEnt baseline trained on DS labels without application of MinCut; (3) the base classifier using MinCut (DSlabels+MinCut) as described above.

Conditions 4-8 train supervised classifiers based on the labels from DSlabels+MinCut: (4) MaxEnt with named entities (NE); (5) MaxEnt with NE and semantic (SEM) features; (6) CRF with NE; (7) MaxEnt with NE and sequential (SQ) features; (8) MaxEnt with NE, SQ, and SEM.

We test statistical significance using the approximate randomization test (Noreen, 1989) on documents with 10,000 iterations at $p < .05$. We achieve classification results above baseline using the MinCut base classifier (line 3) and a considerable improvement through distant supervision. We found that all classifiers using DS labels and Mincut are significantly better than MaxEnt trained on purely rule-based DS labels (line 2). Also, the MaxEnt models using SQ features (lines 7,8) are significantly better than the MinCut base classifier (line 3). For comparison to a chain-based sequence model, we train a CRF (line 6); however, the improvement over MaxEnt (line 4) is not significant.

We found that both semantic (lines 5,8) and sequential (lines 7,8) features help to improve the classifier. The best model (line 7) performs better than MinCut (3) by 3.1% and better than training on purely rule-generated DS labels (line 2) by 5.8%. However, we did not find a cumulative effect (line 8) of the two feature sets.

Generally, the quality of NER is crucial in this task. While IMDb is in general a thoroughly compiled database, it is not perfect. For example, all main characters in *Groundhog Day* are listed with their first name only even though the full names are given in the movie. Also, some entries are intentionally incomplete to avoid spoiling the plot. The data also contains ambiguities between characters and titles (e.g., *Forrest Gump*) that are impossible to resolve with our maximum likelihood method. In some types of movies, e.g., documentaries, the distinction between characters and actors makes little sense. Furthermore, ambiguities like occurrences of common names such as *John* are impossible to resolve if there is no earlier full referring expression (e.g., *John Williams*).

Feature analysis for the best model using DS labels (7) shows that NE features are dominant. This correlation is not surprising as the seed labels were induced based on NE features. Interestingly, some subjective features, e.g., *horrible* have high weights for S-nonrelevance, as they are associated with non-relevant content such as plot descriptions.

To summarize, the results of our experiments using distant supervision show that a sentiment relevance classifier can be trained successfully by labeling data with a few simple feature rules, with

MinCut-based input significantly outperforming the baseline. Named entity recognition, accomplished with data extracted from a domain-specific database, plays a significant rule in creating an initial labeling.

7 Transfer Learning

To address the problem that we do not have enough labeled SR data we now investigate a second semi-supervised method for SR classification, transfer learning (TL). We will use the P&L data (introduced in Section 2.2) for training. This data set has labels that are intended to be subjectivity labels. However, they were automatically created using heuristics and the resulting labels can be either viewed as noisy SR labels or noisy subjectivity labels. Compared to distant supervision, the key advantage of training on P&L is that the training set is much larger, containing around 7 times as much data.

In TL, the key to success is to find a generalized feature representation that supports knowledge transfer. We use a semantic feature generalization method that relies on taxonomies to introduce such features.

We again use MinCut to impose discourse constraints. This time, we first classify the data using a supervised classifier and then use MinCut to smooth the sequences. The baseline (BL) uses a simple bag-of-words representation of sentences for classification which we then extend with semantic features.

7.1 MinCut Setup

We again implement the basic MinCut setup from Section 4. We set the individual weight $\text{ind}(s, x)$ on the edge between sentence s and class x to the estimate $p(x|s)$ returned by the supervised classifier. The parameter c of the MinCut model is tuned using the run count method described in Section 4.

7.2 Experiments and Results

As we would expect, the baseline performance of the supervised classifier on SR is low: 69.9% (Table 4, line 1). MinCut significantly boosts the performance by 7.9% to 77.5% (line 1), a result similar to (Pang and Lee, 2004). Adding semantic features improves supervised classification significantly by 5.7% (75.6% on line 4). When MinCut and both types of semantic features are used together, these improvements are partially cumulative: an improvement over the baseline by 12.6% to 82.5% (line 4).

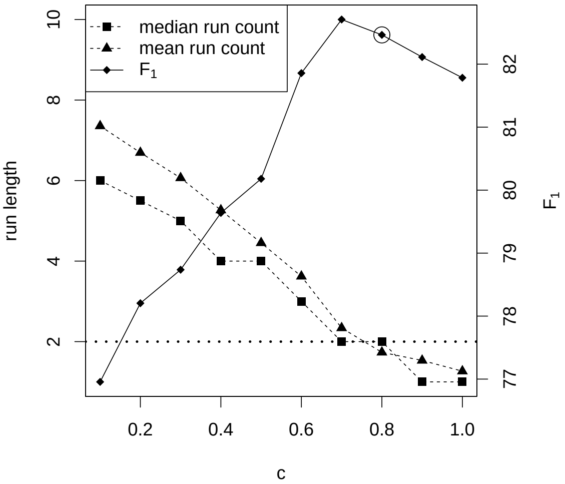

Figure 2: F_1 measure for different values of c. Horizontal line: optimal median run count. Circle: selected point.

We also experiment with a training set where an artificial class imbalance is introduced, matching the 80:20 imbalance of SR:SNR in the S-relevance corpus. After applying MinCut, we find that while the results for BL with and without imbalances does not differ significantly. However, models using CX and VN features and imbalances are actually significantly inferior to the respective balanced versions. This result suggests that MinCut is more effective at coping with class imbalances than artificial balancing.

MinCut and semantic features are successful for TL because both impose constraints that are more useful in a setup where noise is a major problem. MinCut can exploit test set information without supervision as the MinCut graph is built directly on each test set review. If high-confidence information is "seeded" within a document and then spread to neighbors, mistakes with low confidence are corrected. This way, MinCut also leads to a compensation of different class imbalances.

The results are evidence that semantic features are robust to the differences between subjectivity and S-relevance (cf. Section 2). In the CX+VN model, meaningful feature classes receive high weights, e.g., the *human* class from CoreLex which contains professions that are frequently associated with non-relevant plot descriptions.

To illustrate the run-based parameter optimization criterion, we show F_1 and median/mean run lengths for different values of c for the best TL

	Model	base classifier			MinCut		
		F_{SR}	F_{SNR}	F_m	F_{SR}	F_{SNR}	F_m
1	BL	81.1	58.6	69.9	87.2	67.8	77.5^{B}
2	CX	82.9	60.1	71.5^{B}	89.0	70.3	79.7^{BM}
3	VN	85.6	62.1	73.9^{B}	91.4	73.6	82.5^{BM}
4	CX+VN	88.3	62.9	75.6^{B}	92.7	72.2	82.5^{BM}

Table 4: Classification results: F_{SR} (S-relevant F_1), F_{SNR} (S-nonrelevant F_1), and F_m (macro-averaged F_1). $^{\text{B}}$ indicates a significant improvement over the BL base classifier (69.9), $^{\text{M}}$ over BL MinCut (77.5).

setting (line 4) in Figure 2. Due to differences in the base classifier, the optimum of c may vary between the experiments. A weaker base classifier may yield a higher weight on the sequence model, resulting in a larger c. The circled point shows the data point selected through optimization. The optimization criterion does not always correlate perfectly with F_1. However, we find no statistically significant difference between the selected result and the highest F_1 value.

These experiments demonstrate that S-relevance classification improves considerably through TL if semantic feature generalization and unsupervised sequence classification through MinCut are applied.

8 Conclusion

A number of different notions, including subjectivity, have been proposed for distinguishing parts of documents that convey sentiment from those that do not. We introduced *sentiment relevance* to make this distinction and argued that it better reflects the requirements of sentiment analysis systems. Our experiments demonstrated that sentiment relevance and subjectivity are related, but different. To enable other researchers to use this new notion of S-relevance, we have published the annotated S-relevance corpus used in this paper.

Since a large labeled sentiment relevance resource does not yet exist, we investigated semi-supervised approaches to S-relevance classification that do not require S-relevance-labeled data. We showed that a combination of different techniques gives us the best results: semantic generalization features, imposing discourse constraints implemented as the minimum cut graph-theoretic method, automatic “distant” labeling based on a domain-specific metadata database and transfer learning to exploit existing labels for a related classification problem.

In future work, we plan to use sentiment relevance in a downstream task such as review summarization.

Acknowledgments

This work was funded by the DFG through the Sonderforschungsbereich 732. We thank Charles Jochim, Wiltrud Kessler, and Khalid Al Khatib for many helpful comments and discussions.

References

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China, August. Coling 2010 Organizing Committee.

P. Buitelaar. 1998. *CoreLex: systematic polysemy and underspecification*. Ph.D. thesis, Brandeis University.

B. Cherkassky and A. Goldberg. 1995. On implementing push-relabel method for the maximum flow problem. *Integer Programming and Combinatorial Optimization*, pages 157–171.

X. Ding, B. Liu, and P. S. Yu. 2008. A holistic lexicon-based approach to opinion mining. In *WSDM 2008*, pages 231–240.

K. Eguchi and V. Lavrenko. 2006. Sentiment retrieval using generative models. In *EMNLP 2006*, pages 345–354.

L.R. Ford and D.R. Fulkerson. 1956. Maximal flow through a network. *Canadian Journal of Mathematics*, 8(3):399–404.

K. Kipper, A. Korhonen, N. Ryant, and M. Palmer. 2008. A large-scale classification of English verbs. *Language Resources and Evaluation*, 42(1):21–40.

B. Liu. 2010. Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, pages 978–1420085921.

C. Manning and D. Klein. 2003. Optimization, maxent models, and conditional estimation without magic. In *NAACL-HLT 2003: Tutorials*, page 8.

A.K. McCallum. 2002. Mallet: A machine learning for language toolkit.

G.A. Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

E.W. Noreen. 1989. *Computer Intensive Methods for Hypothesis Testing: An Introduction*. Wiley.

B. Pang and L. Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *ACL 2004*, pages 271–278.

B. Pang, L. Lee, and S. Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *ACL-EMNLP 2002*, pages 79–86.

A.M. Popescu and O. Etzioni. 2005. Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 339–346. Association for Computational Linguistics.

A. Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Proceedings of the conference on empirical methods in natural language processing*, volume 1, pages 133–142.

E. Riloff and J. Wiebe. 2003. Learning extraction patterns for subjective expressions. In *EMNLP 2003*, pages 105–112.

M. Taboada, J. Brooke, and M. Stede. 2009. Genre-based paragraph classification for sentiment analysis. In *SIGdial 2009*, pages 62–70.

O. Täckström and R. McDonald. 2011. Discovering fine-grained sentiment with latent variable structured prediction models. In *ECIR 2011*, pages 368–374.

S. Tan and X. Cheng. 2009. Improving SCL model for sentiment-transfer learning. In *ACL 2009*, pages 181–184.

S. Thrun. 1996. Is learning the n-th thing any easier than learning the first? In *NIPS 1996*, pages 640–646.

T. Wilson and J. Wiebe. 2003. Annotating opinions in the world press. In *4th SIGdial Workshop on Discourse and Dialogue*, pages 13–22.

A. Yessenalina, Y. Yue, and C. Cardie. 2010. Multi-level structured models for document-level sentiment classification. In *EMNLP 2010*, pages 1046–1056.

O. Zaidan, J. Eisner, and C. Piatko. 2007. Using annotator rationales to improve machine learning for text categorization. In *NAACL-HLT 2007*, pages 260–267.

L. Zhuang, F. Jing, and X. Zhu. 2006. Movie review mining and summarization. In *CIKM 2006*, pages 43–50.