Multiple Aspect Ranking using the Good Grief Algorithm

Benjamin Snyder and Regina Barzilay
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
{bsnyder,regina}@csail.mit.edu

Abstract

We address the problem of analyzing multiple related opinions in a text. For instance, in a restaurant review such opinions may include food, ambience and service. We formulate this task as a multiple aspect ranking problem, where the goal is to produce a set of numerical scores, one for each aspect. We present an algorithm that jointly learns ranking models for individual aspects by modeling the dependencies between assigned ranks. This algorithm guides the prediction of individual rankers by analyzing meta-relations between opinions, such as agreement and contrast. We prove that our agreement-based joint model is more expressive than individual ranking models. Our empirical results further confirm the strength of the model: the algorithm provides significant improvement over both individual rankers and a state-of-the-art joint ranking model.

1 Introduction

Previous work on sentiment categorization makes an implicit assumption that a single score can express the polarity of an opinion text (Pang et al., 2002; Turney, 2002; Yu and Hatzivassiloglou, 2003). However, multiple opinions on related matters are often intertwined throughout a text. For example, a restaurant review may express judgment on food quality as well as the service and ambience of the restaurant. Rather than lumping these aspects into a single score, we would like to capture each aspect of the writer's opinion separately, thereby providing a more fine-grained view of opinions in the review.

To this end, we aim to predict a set of numeric ranks that reflects the user's satisfaction for each aspect. In the example above, we would assign a numeric rank from 1-5 for each of: food quality, service, and ambience.

A straightforward approach to this task would be to rank[1] the text independently for each aspect, using standard ranking techniques such as regression or classification. However, this approach fails to exploit meaningful dependencies between users' judgments across different aspects. Knowledge of these dependencies can be crucial in predicting accurate ranks, as a user's opinions on one aspect can influence his or her opinions on others.

The algorithm presented in this paper models the dependencies between different labels via *the agreement relation*. The agreement relation captures whether the user equally likes all aspects of the item or whether he or she expresses different degrees of satisfaction. Since this relation can often be determined automatically for a given text (Marcu and Echihabi, 2002), we can readily use it to improve rank prediction.

The Good Grief model consists of a ranking model for each aspect as well as an agreement model which predicts whether or not all rank aspects are

[1] In this paper, *ranking* refers to the task of assigning an integer from 1 to k to each instance. This task is sometimes referred to as "ordinal regression" (Crammer and Singer, 2001) and "rating prediction" (Pang and Lee, 2005).

Proceedings of NAACL HLT 2007, pages 300–307,
Rochester, NY, April 2007.

equal. The Good Grief decoding algorithm predicts a set of ranks – one for each aspect – which maximally satisfy the preferences of the individual rankers and the agreement model. For example, if the agreement model predicts consensus but the individual rankers select ranks $\langle 5, 5, 4 \rangle$, then the decoder decides whether to trust the the third ranker, or alter its prediction and output $\langle 5, 5, 5 \rangle$ to be consistent with the agreement prediction. To obtain a model well-suited for this decoding, we also develop a joint training method that conjoins the training of multiple aspect models.

We demonstrate that the agreement-based joint model is more expressive than individual ranking models. That is, every training set that can be perfectly ranked by individual ranking models for each aspect can also be perfectly ranked with our joint model. In addition, we give a simple example of a training set which cannot be perfectly ranked without agreement-based joint inference. Our experimental results further confirm the strength of the Good Grief model. Our model significantly outperforms individual ranking models as well as a state-of-the-art joint ranking model.

2 Related Work

Sentiment Classification Traditionally, categorization of opinion texts has been cast as a binary classification task (Pang et al., 2002; Turney, 2002; Yu and Hatzivassiloglou, 2003; Dave et al., 2003). More recent work (Pang and Lee, 2005; Goldberg and Zhu, 2006) has expanded this analysis to the ranking framework where the goal is to assess review polarity on a multi-point scale. While this approach provides a richer representation of a single opinion, it still operates on the assumption of one opinion per text. Our work generalizes this setting to the problem of analyzing multiple opinions – or multiple aspects of an opinion. Since multiple opinions in a single text are related, it is insufficient to treat them as separate single-aspect ranking tasks. This motivates our exploration of a new method for joint multiple aspect ranking.

Ranking The ranking, or ordinal regression, problem has been extensivly studied in the Machine Learning and Information Retrieval communities. In this section we focus on two online ranking methods which form the basis of our approach. The first is a model proposed by Crammer and Singer (2001). The task is to predict a rank $y \in \{1, ..., k\}$ for every input $\mathbf{x} \in \mathbb{R}^n$. Their model stores a weight vector $\mathbf{w} \in \mathbb{R}^n$ and a vector of increasing boundaries $b_0 = -\infty \leq b_1 \leq ... \leq b_{k-1} \leq b_k = \infty$ which divide the real line into k segments, one for each possible rank. The model first scores each input with the weight vector: $score(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}$. Finally, the model locates $score(\mathbf{x})$ on the real line and returns the appropriate rank as indicated by the boundaries. Formally, the model returns the rank r such that $b_{r-1} \leq score(\mathbf{x}) < b_r$. The model is trained with the Perceptron Ranking algorithm (or "PRank algorithm"), which reacts to incorrect predictions on the training set by updating the weight and boundary vectors. The PRanking model and algorithm were tested on the `EachMovie` dataset with a separate ranking model learned for each user in the database.

An extension of this model is provided by Basilico and Hofmann (2004) in the context of collaborative filtering. Instead of training a separate model for each user, Basilico and Hofmann train a joint ranking model which shares a set of boundaries across all users. In addition to these shared boundaries, user-specific weight vectors are stored. To compute the score for input $\mathbf{x}$ and user i, the weight vectors for *all* users are employed:

$$score_i(\mathbf{x}) = \mathbf{w}[i] \cdot \mathbf{x} + \sum_j sim(i, j)(\mathbf{w}[j] \cdot \mathbf{x}) \quad (1)$$

where $0 \leq sim(i, j) \leq 1$ is the cosine similarity between users i and j, computed on the entire training set. Once the score has been computed, the prediction rule follows that of the PRanking model. The model is trained using the PRank algorithm, with the exception of the new definition for the scoring function.[2] While this model shares information between the different ranking problems, it fails to explicitly model relations between the rank predictions. In contrast, our algorithm uses an agreement model to learn such relations and inform joint predictions.

[2] In the notation of Basilico and Hofmann (2004), this definition of $score_i(\mathbf{x})$ corresponds to the kernel $K = (K_U^{id} + K_U^{co}) \oplus K_X^{at}$.

3 The Algorithm

The goal of our algorithm is to find a rank assignment that is consistent with predictions of individual rankers and the agreement model. To this end, we develop the Good Grief decoding procedure that minimizes the dissatisfaction (*grief*) of individual components with a joint prediction. In this section, we formally define the grief of each component, and a mechanism for its minimization. We then describe our method for joint training of individual rankers that takes into account the Good Grief decoding procedure.

3.1 Problem Formulation

In an *m-aspect ranking problem*, we are given a training sequence of instance-label pairs $(\mathbf{x}^1, \mathbf{y}^1), ..., (\mathbf{x}^t, \mathbf{y}^t), ...$ Each instance $\mathbf{x}^t$ is a feature vector in $\mathbb{R}^n$ and the label $\mathbf{y}^t$ is a vector of m ranks in $\mathcal{Y}^m$, where $\mathcal{Y} = \{1, .., k\}$ is the set of possible ranks. The i^{th} component of $\mathbf{y}^t$ is the rank for the i^{th} aspect, and will be denoted by $y[i]^t$. The goal is to learn a mapping from instances to rank sets, $H : \mathcal{X} \rightarrow \mathcal{Y}^m$, which minimizes the distance between predicted ranks and true ranks.

3.2 The Model

Our *m-aspect ranking model* contains $m+1$ components: $(\langle \mathbf{w}[1], \mathbf{b}[1] \rangle, ..., \langle \mathbf{w}[m], \mathbf{b}[m] \rangle, \mathbf{a})$. The first m components are individual ranking models, one for each aspect, and the final component is the agreement model. For each aspect $i \in 1...m$, $\mathbf{w}[i] \in \mathbb{R}^n$ is a vector of weights on the input features, and $\mathbf{b}[i] \in \mathbb{R}^{k-1}$ is a vector of boundaries which divide the real line into k intervals, corresponding to the k possible ranks. The default prediction of the aspect ranking model simply uses the ranking rule of the PRank algorithm. This rule predicts the rank r such that $b[i]_{r-1} \leq score_i(\mathbf{x}) < b[i]_r$.[3] The value $score_i(\mathbf{x})$ can be defined simply as the dot product $\mathbf{w}[i] \cdot \mathbf{x}$, or it can take into account the weight vectors for other aspects weighted by a measure of inter-aspect similarity. We adopt the definition given in equation 1, replacing the user-specific weight vectors with our aspect-specific weight vectors.

The agreement model is a vector of weights $\mathbf{a} \in \mathbb{R}^n$. A value of $\mathbf{a} \cdot \mathbf{x} > 0$ predicts that the ranks of all m aspects are equal, and a value of $\mathbf{a} \cdot \mathbf{x} \leq 0$ indicates disagreement. The absolute value $|\mathbf{a} \cdot \mathbf{x}|$ indicates the confidence in the agreement prediction.

The goal of the decoding procedure is to predict a joint rank for the m aspects which satisfies the individual ranking models as well as the agreement model. For a given input $\mathbf{x}$, the individual model for aspect i predicts a default rank $\hat{y}[i]$ based on its feature weight and boundary vectors $\langle \mathbf{w}[i], \mathbf{b}[i] \rangle$. In addition, the agreement model makes a prediction regarding rank consensus based on $\mathbf{a} \cdot \mathbf{x}$. However, the default aspect predictions $\hat{y}[1] \ldots \hat{y}[m]$ may not accord with the agreement model. For example, if $\mathbf{a} \cdot \mathbf{x} > 0$, but $\hat{y}[i] \neq \hat{y}[j]$ for some $i, j \in 1...m$, then the agreement model predicts complete consensus, whereas the individual aspect models do not.

We therefore adopt a joint prediction criterion which simultaneously takes into account *all* model components – individual aspect models as well as the agreement model. For each possible prediction $\mathbf{r} = (r[1], ..., r[m])$ this criterion assesses the level of *grief* associated with the i^{th}-aspect ranking model, $g_i(\mathbf{x}, r[i])$. Similarly, we compute the grief of the agreement model with the joint prediction, $g_a(\mathbf{x}, \mathbf{r})$ (both g_i and g_a are defined formally below). The decoder then predicts the m ranks which minimize the overall grief:

$$H(\mathbf{x}) = \arg \min_{\mathbf{r} \in \mathcal{Y}^m} \left[g_a(\mathbf{x}, \mathbf{r}) + \sum_{i=1}^{m} g_i(\mathbf{x}, r[i]) \right] \quad (2)$$

If the default rank predictions for the aspect models, $\hat{\mathbf{y}} = (\hat{y}[1], ..., \hat{y}[m])$, are in accord with the agreement model (both indicating consensus or both indicating contrast), then the grief of all model components will be zero, and we simply output $\hat{\mathbf{y}}$. On the other hand, if $\hat{\mathbf{y}}$ indicates disagreement but the agreement model predicts consensus, then we have the option of predicting $\hat{\mathbf{y}}$ and bearing the grief of the agreement model. Alternatively, we can predict some consensus $\mathbf{y}'$ (i.e. with $y'[i] = y'[j], \forall i, j$) and bear the grief of the component ranking models. The decoder H chooses the option with lowest overall grief.[4]

[3] More precisely (taking into account the possibility of ties): $\hat{y}[i] = \min_{r \in \{1,..,k\}} \{r : score_i(\mathbf{x}) - b[i]_r < 0\}$

[4] This decoding criterion assumes that the griefs of the com-

Now we formally define the measures of *grief* used in this criterion.

Aspect Model Grief We define the grief of the i^{th}-aspect ranking model with respect to a rank r to be the smallest magnitude correction term which places the input's score into the r^{th} segment of the real line:

$$g_i(\mathbf{x}, r) = \min |c|$$
$$\text{s.t.}$$
$$b[i]_{r-1} \leq score_i(\mathbf{x}) + c < b[i]_r$$

Agreement Model Grief Similarly, we define the grief of the agreement model with respect to a joint rank $\mathbf{r} = (r[1], \ldots, r[m])$ as the smallest correction needed to bring the agreement score into accord with the agreement relation between the individual ranks $r[1], \ldots, r[m]$:

$$g_a(\mathbf{x}, \mathbf{r}) = \min |c|$$
$$\text{s.t.}$$
$$\mathbf{a} \cdot \mathbf{x} + c > 0 \wedge \forall i, j \in 1...m : r[i] = r[j]$$
$$\vee$$
$$\mathbf{a} \cdot \mathbf{x} + c \leq 0 \wedge \exists i, j \in 1...m : r[i] \neq r[j]$$

3.3 Training

Ranking models Pseudo-code for Good Grief training is shown in Figure 1. This training algorithm is based on PRanking (Crammer and Singer, 2001), an online perceptron algorithm. The training is performed by iteratively ranking each training input $\mathbf{x}$ and updating the model. If the predicted rank $\hat{y}$ is equal to the true rank y, the weight and boundaries vectors remain unchanged. On the other hand, if $\hat{y} \neq y$, then the weights and boundaries are updated to improve the prediction for $\mathbf{x}$ (step 4.c in Figure 1). See (Crammer and Singer, 2001) for explanation and analysis of this update rule.

Our algorithm departs from PRanking by conjoining the updates for the m ranking models. We achieve this by using Good Grief decoding at each step throughout training. Our decoder $H(\mathbf{x})$ (from equation 2) uses *all* the aspect component models as well as the (previously trained) agreement model to determine the predicted rank for each aspect. In concrete terms, for every training instance $\mathbf{x}$, we predict the ranks of all aspects simultaneously (step 2 in Figure 1). Then, for each aspect we make a separate update based on this joint prediction (step 4 in Figure 1), instead of using the individual models' predictions.

Agreement model The agreement model $\mathbf{a}$ is assumed to have been previously trained on the same training data. An instance is labeled with a positive label if all the ranks associated with this instance are equal. The rest of the instances are labeled as negative. This model can use any standard training algorithm for binary classification such as Perceptron or SVM optimization.

3.4 Feature Representation

Ranking Models Following previous work on sentiment classification (Pang et al., 2002), we represent each review as a vector of lexical features. More specifically, we extract all unigrams and bigrams, discarding those that appear fewer than three times. This process yields about 30,000 features.

Agreement Model The agreement model also operates over lexicalized features. The effectiveness of these features for recognition of discourse relations has been previously shown by Marcu and Echihabi (2002). In addition to unigrams and bigrams, we also introduce a feature that measures the maximum contrastive distance between pairs of words in a review. For example, the presence of *"delicious"* and *"dirty"* indicate high contrast, whereas the pair *"expensive"* and *"slow"* indicate low contrast. The contrastive distance for a pair of words is computed by considering the difference in relative weight assigned to the words in individually trained PRanking models.

4 Analysis

In this section, we prove that our model is able to perfectly rank a strict superset of the training corpora perfectly rankable by m ranking models individually. We first show that if the independent ranking models can individually rank a training set perfectly, then our model can do so as well. Next, we show that our model is more expressive by providing

ponent models are comparable. In practice, we take an uncalibrated agreement model $\mathbf{a}'$ and reweight it with a tuning parameter: $\mathbf{a} = \alpha\mathbf{a}'$. The value of α is estimated using the development set. We assume that the griefs of the ranking models are comparable since they are jointly trained.

Input : $(\mathbf{x}^1, \mathbf{y}^1), ..., (\mathbf{x}^T, \mathbf{y}^T)$, Agreement model $\mathbf{a}$, Decoder defintion $H(\mathbf{x})$ (from equation 2).
Initialize : Set $\mathbf{w}[i]^1 = 0$, $b[i]^1_1, ..., b[i]^1_{k-1} = 0$, $b[i]^1_k = \infty$, $\forall i \in 1...m$.
Loop : For $t = 1, 2, ..., T$:

1. Get a new instance $\mathbf{x}^t \in \mathbb{R}^n$.
2. Predict $\hat{\mathbf{y}}^t = H(\mathbf{x}; \mathbf{w}^t, \mathbf{b}^t, \mathbf{a})$ (Equation 2).
3. Get a new label $\mathbf{y}^t$.
4. For aspect $i = 1, ..., m$:

If $\hat{y}[i]^t \neq y[i]^t$ update model (otherwise set $\mathbf{w}[i]^{t+1} = \mathbf{w}[i]^t,\ b[i]^{t+1}_r = b[i]^t_r,\ \forall r$):

4.a For $r = 1, ..., k-1$: If $y[i]^t \leq r$ then $y[i]^t_r = -1$
else $y[i]^t_r = 1$.

4.b For $r = 1, ..., k-1$: If $(\hat{y}[i]^t - r) y[i]^t_r \leq 0$ then $\tau[i]^t_r = y[i]^t_r$
else $\tau[i]^t_r = 0$.

4.c Update $\mathbf{w}[i]^{t+1} \leftarrow \mathbf{w}[i]^t + (\sum_r \tau[i]^t_r)\mathbf{x}^t$.
For $r = 1, ..., k-1$ update : $b[i]^{t+1}_r \leftarrow b[i]^t_r - \tau[i]^t_r$.

Output : $H(\mathbf{x}; \mathbf{w}^{T+1}, \mathbf{b}^{T+1}, \mathbf{a})$.

Figure 1: Good Grief Training. The algorithm is based on PRanking training algorithm. Our algorithm differs in the joint computation of all aspect predictions $\hat{\mathbf{y}}^t$ based on the Good Grief Criterion (step 2) and the calculation of updates for each aspect based on the joint prediction (step 4).

a simple illustrative example of a training set which can only be perfectly ranked with the inclusion of an agreement model.

First we introduce some notation. For each training instance $(\mathbf{x}^t, \mathbf{y}^t)$, each aspect $i \in 1...m$, and each rank $r \in 1...k$, define an auxiliary variable $y[i]^t_r$ with $y[i]^t_r = -1$ if $y[i]^t \leq r$ and $y[i]^t_r = 1$ if $y[i]^t > r$. In words, $y[i]^t_r$ indicates whether the *true* rank $y[i]^t$ is to the right or left of a *potential* rank r.

Now suppose that a training set $(\mathbf{x}^1, \mathbf{y}^1), ..., (\mathbf{x}^T, \mathbf{y}^T)$ is perfectly rankable for each aspect independently. That is, for each aspect $i \in 1...m$, there exists some ideal model $v[i]^* = (w[i]^*, b[i]^*)$ such that the signed distance from the prediction to the r^{th} boundary: $\mathbf{w}[i]^* \cdot \mathbf{x}^t - b[i]^*_r$ has the same sign as the auxiliary variable $y[i]^t_r$. In other words, the minimum margin over all training instances and ranks, $\gamma = \min_{r,t}\{(\mathbf{w}[i]^* \cdot \mathbf{x}^t - b[i]^*_r) y[i]^t_r\}$, is no less than zero.

Now for the t^{th} training instance, define an agreement auxiliary variable a^t, where $a^t = 1$ when all aspects agree in rank and $a^t = -1$ when at least two aspects disagree in rank. First consider the case where the agreement model $\mathbf{a}$ perfectly classifies all training instances: $(\mathbf{a} \cdot \mathbf{x}^t)a^t > 0, \forall t$. It is clear that Good Grief decoding with the ideal joint model $(\langle \mathbf{w}[1]^*, \mathbf{b}[1]^* \rangle, ..., \langle \mathbf{w}[m]^*, \mathbf{b}[m]^* \rangle, \mathbf{a})$ will produce the same output as the component ranking models run separately (since the grief will always be zero for the default rank predictions). Now consider the case where the training data is not linearly separable with regard to agreement classification. Define the margin of the worst case error to be $\beta = \max_t\{|(\mathbf{a} \cdot \mathbf{x}^t)| : (\mathbf{a} \cdot \mathbf{x}^t)a^t < 0\}$. If $\beta < \gamma$, then again Good Grief decoding will always produce the default results (since the grief of the agreement model will be at most β in cases of error, whereas the grief of the ranking models for any deviation from their default predictions will be at least γ). On the other hand, if $\beta \geq \gamma$, then the agreement model errors could potentially disrupt the perfect ranking. However, we need only rescale $w^* := w^*(\frac{\beta}{\gamma} + \epsilon)$ and $b^* := b^*(\frac{\beta}{\gamma} + \epsilon)$ to ensure that the grief of the ranking models will always exceed the grief of the agreement model in cases where the latter is in error. Thus whenever independent ranking models can perfectly rank a training set, a joint ranking model with Good Grief decoding can do so as well.

Now we give a simple example of a training set which can only be perfectly ranked with the addition of an agreement model. Consider a training set of four instances with two rank aspects:

$\langle \mathbf{x}^1, \mathbf{y}^1 \rangle = \langle (1,0,1), \quad (2,1) \rangle$
$\langle \mathbf{x}^2, \mathbf{y}^2 \rangle = \langle (1,0,0), \quad (2,2) \rangle$
$\langle \mathbf{x}^3, \mathbf{y}^3 \rangle = \langle (0,1,1), \quad (1,2) \rangle$
$\langle \mathbf{x}^4, \mathbf{y}^4 \rangle = \langle (0,1,0), \quad (1,1) \rangle$

We can interpret these inputs as feature vectors corresponding to the presence of "good", "bad", and "but not" in the following four sentences:

The food was **good**, **but not** the ambience.

The food was **good**, and so was the ambience.

The food was **bad**, **but not** the ambience.

The food was **bad**, and so was the ambience.

We can further interpret the first rank aspect as the quality of food, and the second as the quality of the ambience, both on a scale of 1-2.

A simple ranking model which only considers the words "good" and "bad" perfectly ranks the food aspect. However, it is easy to see that no single model perfectly ranks the ambience aspect. Consider any model $\langle \mathbf{w}, \mathbf{b} = (b) \rangle$. Note that $\mathbf{w} \cdot \mathbf{x}^1 < b$ and $\mathbf{w} \cdot \mathbf{x}^2 \geq b$ together imply that $w_3 < 0$, whereas $\mathbf{w} \cdot \mathbf{x}^3 \geq b$ and $\mathbf{w} \cdot \mathbf{x}^4 < b$ together imply that $w_3 > 0$. Thus independent ranking models cannot perfectly rank this corpus.

The addition of an agreement model, however, can easily yield a perfect ranking. With $\mathbf{a} = (0, 0, -5)$ (which predicts contrast with the presence of the words "but not") and a ranking model for the ambience aspect such as $\mathbf{w} = (1, -1, 0), \mathbf{b} = (0)$, the Good Grief decoder will produce a perfect rank.

5 Experimental Set-Up

We evaluate our multi-aspect ranking algorithm on a corpus[5] of restaurant reviews available on the website `http://www.we8there.com`. Reviews from this website have been previously used in other sentiment analysis tasks (Higashinaka et al., 2006). Each review is accompanied by a set of five ranks, each on a scale of 1-5, covering food, ambience, service, value, and overall experience. These ranks are provided by consumers who wrote original reviews. Our corpus does not contain incomplete data points since all the reviews available on this website contain both a review text and the values for all the five aspects.

Training and Testing Division Our corpus contains 4,488 reviews, averaging 115 words. We randomly select 3,488 reviews for training, 500 for development and 500 for testing.

Parameter Tuning We used the development set to determine optimal numbers of training iterations for our model and for the baseline models. Also, given an initial uncalibrated agreement model $\mathbf{a}'$, we define our agreement model to be $\mathbf{a} = \alpha \mathbf{a}'$ for an appropriate scaling factor α. We tune the value of α on the development set.

Corpus Statistics Our training corpus contains 528 among $5^5 = 3025$ possible rank sets. The most frequent rank set $\langle 5, 5, 5, 5, 5 \rangle$ accounts for 30.5% of the training set. However, no other rank set comprises more than 5% of the data. To cover 90% of occurrences in the training set, 227 rank sets are required. Therefore, treating a rank tuple as a single label is not a viable option for this task. We also find that reviews with full agreement across rank aspects are quite common in our corpus, accounting for 38% of the training data. Thus an agreement-based approach is natural and relevant.

A rank of 5 is the most common rank for all aspects and thus a prediction of all 5's gives a MAJORITY baseline and a natural indication of task difficulty.

Evaluation Measures We evaluate our algorithm and the baseline using *ranking loss* (Crammer and Singer, 2001; Basilico and Hofmann, 2004). Ranking loss measures the average distance between the true rank and the predicted rank. Formally, given N test instances $(\mathbf{x}^1, \mathbf{y}^1), ..., (\mathbf{x}^N, \mathbf{y}^N)$ of an m-aspect ranking problem and the corresponding predictions $\hat{\mathbf{y}}^1, ..., \hat{\mathbf{y}}^N$, ranking loss is defined as $\sum_{t,i} \frac{|y[i]^t - \hat{y}[i]^t|}{mN}$. Lower values of this measure correspond to a better performance of the algorithm.

6 Results

Comparison with Baselines Table 1 shows the performance of the Good Grief training algorithm GG TRAIN+DECODE along with various baselines, including the simple MAJORITY baseline mentioned in section 5. The first competitive baseline, PRANK, learns a separate ranker for each aspect using the PRank algorithm. The second competitive baseline, SIM, shares the weight vectors across aspects using a similarity measure (Basilico and Hofmann, 2004).

[5] Data and code used in this paper are available at `http://people.csail.mit.edu/bsnyder/naacl07`

	Food	Service	Value	Atmosphere	Experience	**Total**
MAJORITY	0.848	1.056	1.030	1.044	1.028	1.001
PRANK	0.606	0.676	0.700	0.776	0.618	0.675
SIM	0.562	0.648	0.706	0.798	0.600	0.663
GG DECODE	0.544	0.648	0.704	0.798	0.584	0.656
GG TRAIN+DECODE	**0.534**	**0.622**	**0.644**	**0.774**	**0.584**	**0.632**
GG ORACLE	0.510	0.578	0.674	0.694	0.518	0.595

Table 1: Ranking loss on the test set for variants of Good Grief and various baselines.

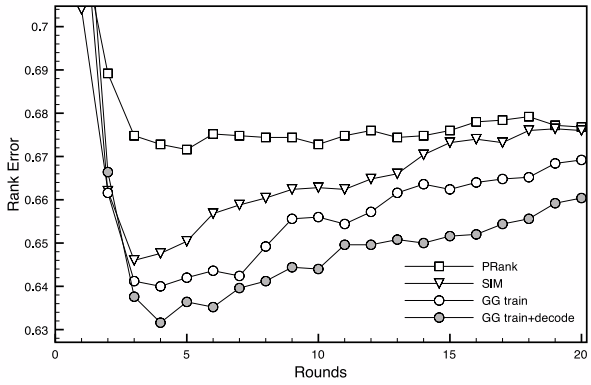

Figure 2: Rank loss for our algorithm and baselines as a function of training round.

Both of these methods are described in detail in Section 2. In addition, we consider two variants of our algorithm: GG DECODE employs the PRank training algorithm to independently train all component ranking models and only applies Good Grief decoding at test time. GG ORACLE uses Good Grief training and decoding but in both cases is given perfect knowledge of whether or not the true ranks all agree (instead of using the trained agreement model).

Our model achieves a rank error of 0.632, compared to 0.675 for PRANK and 0.663 for SIM. Both of these differences are statistically significant at $p < 0.002$ by a Fisher Sign Test. The gain in performance is observed across all five aspects. Our model also yields significant improvement ($p < 0.05$) over the decoding-only variant GG DECODE, confirming the importance of joint training. As shown in Figure 2, our model demonstrates consistent improvement over the baselines across all the training rounds.

Model Analysis We separately analyze our performance on the 210 test instances where all the target ranks agree and the remaining 290 instances where there is some contrast. As Table 2 shows, we outperform the PRANK baseline in both cases. However on the consensus instances we achieve a relative reduction in error of 21.8% compared to only a 1.1% reduction for the other set. In cases of consensus, the agreement model can guide the ranking models by reducing the decision space to five rank sets. In cases of disagreement, however, our model does not provide sufficient constraints as the vast majority of ranking sets remain viable. This explains the performance of GG ORACLE, the variant of our algorithm with perfect knowledge of agreement/disagreement facts. As shown in Table 1, GG ORACLE yields substantial improvement over our algorithm, but most of this gain comes from consensus instances (see Table 2).

	Consensus	Non-consensus
PRANK	0.414	0.864
GG TRAIN+DECODE	0.324	0.854
GG ORACLE	0.281	0.830

Table 2: Ranking loss for our model and PRANK computed separately on cases of actual consensus and actual disagreement.

We also examine the impact of the agreement model accuracy on our algorithm. The agreement model, when considered on its own, achieves classification accuracy of 67% on the test set, compared to a majority baseline of 58%. However, those instances with high confidence $|\mathbf{a} \cdot \mathbf{x}|$ exhibit substantially higher classification accuracy. Figure 3 shows the performance of the agreement model as a function of the confidence value. The 10% of the data with highest confidence values can be classified by

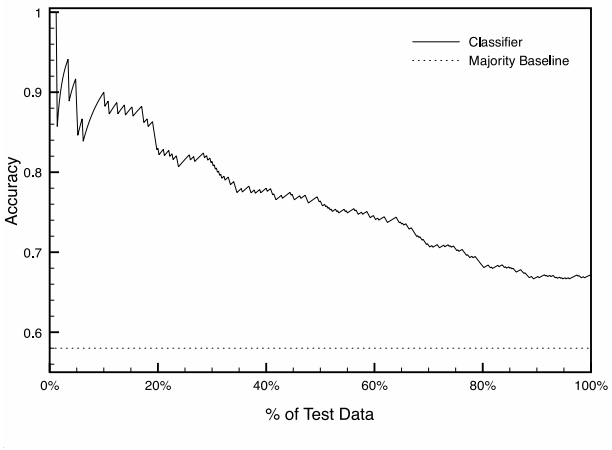

Figure 3: Accuracy of the agreement model on subsets of test instances with highest confidence $|\mathbf{a} \cdot \mathbf{x}|$.

the agreement model with 90% accuracy, and the third of the data with highest confidence can be classified at 80% accuracy.

This property explains why the agreement model helps in joint ranking even though its overall accuracy may seem low. Under the Good Grief criterion, the agreement model's prediction will only be enforced when its grief outweighs that of the ranking models. Thus in cases where the prediction confidence ($|\mathbf{a}\cdot\mathbf{x}|$) is relatively low,[6] the agreement model will essentially be ignored.

7 Conclusion and Future Work

We considered the problem of analyzing multiple related aspects of user reviews. The algorithm presented jointly learns ranking models for individual aspects by modeling the dependencies between assigned ranks. The strength of our algorithm lies in its ability to guide the prediction of individual rankers using rhetorical relations between aspects such as agreement and contrast. Our method yields significant empirical improvements over individual rankers as well as a state-of-the-art joint ranking model.

Our current model employs a single rhetorical relation – agreement vs. contrast – to model dependencies between different opinions. As our analysis shows, this relation does not provide sufficient constraints for non-consensus instances. An avenue for future research is to consider the impact of additional rhetorical relations between aspects. We also plan to theoretically analyze the convergence properties of this and other joint perceptron algorithms.

[6] What counts as "relatively low" will depend on both the value of the tuning parameter α and the confidence of the component ranking models for a particular input $\mathbf{x}$.

Acknowledgments

The authors acknowledge the support of the National Science Foundation (CAREER grant IIS-0448168 and grant IIS-0415865) and the Microsoft Research Faculty Fellowship. Thanks to Michael Collins, Pawan Deshpande, Jacob Eisenstein, Igor Malioutov, Luke Zettlemoyer, and the anonymous reviewers for helpful comments and suggestions. Thanks also to Vasumathi Raman for programming assistance. Any opinions, findings, and conclusions or recommendations expressed above are those of the authors and do not necessarily reflect the views of the NSF.

References

J. Basilico, T. Hofmann. 2004. Unifying collaborative and content-based filtering. In *Proceedings of the ICML*, 65–72.

K. Crammer, Y. Singer. 2001. Pranking with ranking. In *NIPS*, 641–647.

K. Dave, S. Lawrence, D. Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of WWW*, 519–528.

A. B. Goldberg, X. Zhu. 2006. Seeing stars when there aren't many stars: Graph-based semi-supervised learning for sentiment categorization. In *Proceedings of HLT/NAACL workshop on TextGraphs*, 45–52.

R. Higashinaka, R. Prasad, M. Walker. 2006. Learning to generate naturalistic utterances using reviews in spoken dialogue systems. In *Proceedings of COLING/ACL*, 265–272.

D. Marcu, A. Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *Proceedings of ACL*, 368–375.

B. Pang, L. Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*, 115–124.

B. Pang, L. Lee, S. Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP*, 79–86.

P. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classsification of reviews. In *Proceedings of the ACL*, 417–424.

H. Yu, V. Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of EMNLP*, 129–136.