Addressing Class Imbalance for Improved Recognition of Implicit Discourse Relations

Junyi Jessy Li
University of Pennsylvania
ljunyi@seas.upenn.edu

Ani Nenkova
University of Pennsylvania
nenkova@seas.upenn.edu

Abstract

In this paper we address the problem of skewed class distribution in implicit discourse relation recognition. We examine the performance of classifiers for both binary classification predicting if a particular relation holds or not and for multi-class prediction. We review prior work to point out that the problem has been addressed differently for the binary and multi-class problems. We demonstrate that adopting a unified approach can significantly improve the performance of multi-class prediction. We also propose an approach that makes better use of the full annotations in the training set when downsampling is used. We report significant absolute improvements in performance in multi-class prediction, as well as significant improvement of binary classifiers for detecting the presence of implicit Temporal, Comparison and Contingency relations.

1 Introduction

Discourse relations holding between adjacent sentences in text play an essential role in establishing local coherence and contribute to the semantic interpretation of the text. For example, the causal relationship is helpful for textual entailment or question answering while restatement and exemplification are important for automatic summarization.

Predicting the type of implicit relations, which are not signaled by any of the common explicit discourse connectives such as *because*, *however*, has proven to be a most challenging task in discourse analysis. The Penn Discourse Treebank (PDTB) (Prasad et al., 2008) provided valuable annotations of implicit relations. Most research to date has focused on developing and refining lexical and linguistically rich features for the task (Pitler et al., 2009; Lin et al., 2009; Park and Cardie, 2012). Mostly ignored remains the problem of addressing the highly skewed distribution of implicit discourse relations. Only about 35% of pairs of adjacent sentences in the PDTB are connected by three of the four top level discourse relation: 5% participate in *Temporal* relation, 10% in *Comparison* (contrast) and 20% in *Contingency* (causal) relations. The remaining pairs are connected by the catch-all *Expansion* relation (40%) or by some other linguistic devices (24%). Finer grained relations of interest to particular applications account for increasingly smaller percentage of the PDTB data.

Class imbalance is particularly problematic for training a binary classifier to distinguish one relation from the rest. As we will show later, it also impacts the performance of multi-class prediction in which each pair of sentences is labeled with one of the five possible relations.

All prior work has resorted to downsampling the training data for binary classifiers to distinguish a particular relation and use the full training set for multi-class prediction. In this paper we compare several methods for addressing the skewed class distribution during training: downsampling, upsampling and computing feature weights and performing feature selection on the unaltered full training data. A major motivation for our work is to establish if any of the alternatives to downsampling would prove beneficial, because in downsampling most of the expensively annotated data is not used in the model. In addition, we seek to align the treatment of data imbalance for the binary and multi-class tasks. We show that downsampling in general leads to the best prediction accuracy but that the alternative models provide complementary information and significant improvement can be obtained by combining both types of models. We also report significant improvement of multi-class prediction accuracy,

Proceedings of the SIGDIAL 2014 Conference, pages 142–150,
Philadelphia, U.S.A., 18-20 June 2014.

achieved by using the alternative binary classifiers to perform the task.

2 The Penn Discourse Treebank

In the PDTB, discourse relations are viewed as a predicate with two arguments. The predicate is the relation, the arguments correspond to the minimum spans of text whose interpretations are the abstract objects between which the relation holds. Consider the following example of a contrast relation. The italic and bold fonts mark the arguments of the relation.

Commonwealth Edison said *the ruling could force it to slash its 1989 earnings by $1.55 a share*. [Implicit = BY COMPARISON] **For 1988, Commonwealth Edison reported earnings of $737.5 million, or $3.01 a share**.

For explicit relations, the predicate is marked by a discourse connective that occurs in the text, i.e. *because*, *however*, *for example*.

Implicit relations are marked between adjacent sentences in the same paragraph. They are inferred by the reader but are not lexically marked. Alternative lexicalizations (*AltLex*) are the ones where there is a phrase in the sentence implying the relation but the phrase itself was not one of the explicit discourse connectives. There are 16,224 and 624 examples of implicit and *AltLex* relations, respectively, in the PDTB.

The sense of discourse relations in the PDTB is organized in a three-tiered hierarchy. The four top level relations are: *Temporal* (the two arguments are related temporally), *Comparison* (contrast), *Contingency* (causal) and *Expansion* (one argument is the expansion of the other and continues the context) (Miltsakaki et al., 2008). These are the classes we focus on in our work.

Finally, 5,210 pairs of adjacent sentences were marked as related by an entity relation (*EntRel*), by virtue of the repetition of the same entity or topic. *EntRel*s were marked only if no other relation could be identified and they are not considered a discourse relation, rather an alternative discourse phenomena related to entity coherence (Grosz et al., 1995). There are 254 pairs of sentences where no discourse relation was identified (*NoRel*).

Pitler et al. (2008) has shown that performance as high as 93% in accuracy can be easily achieved for the explicit relations, because the connective itself is a highly informative feature. Efforts in identifying the argument spans have also yielded high accuracies (Lin et al., 2014; Elwell and Baldridge, 2008; Ghosh et al., 2011).

However, in the absence of a connective, recognizing non-explicit relations, which includes implicit relations, alternative lexicalizations, entity relation and no relation present, has proven to be a real challenge. Prior work on supervised implicit discourse recognition studied a wide range of features including lexical, syntactic, verb classes, semantic groups via General Inquirer and polarity (Pitler et al., 2009; Lin et al., 2009). Park and Cardie (2012) studied the combination of features and achieved better performance with a different combination for each individual relation. Methods for improving the sparsity of lexical representations have been proposed (Hernault et al., 2010; Biran and McKeown, 2013), as well as web-driven approaches which reduce the problem to explicit relation recognition (Hong et al., 2012).

Remarkably, no prior work has discussed the highly skewed class distribution of discourse relation types. The tacitly adopted solution has been to downsample the negative examples for one-vs-all binary classification aimed at discovering if a particular relation holds and keeping the full training set for multi-class prediction.

To highlight the problem, in Table 1 we show the distribution of implicit relation classes in the entire PDTB. In our work, we aim to develop classifiers to identify the four top-level relations listed in the table[1].

	# of samples	Percentage
Temporal	1038	4.3%
Comparison	2550	11.3%
Contingency	4532	20%
Expansion	9082	40%

Table 1: Distribution of implicit relations in the PDTB.

3 Experimental settings

In our experiments, we used all non-explicit instances in the PDTB sections 2-19 for training and those in sections 20-24 for testing. Like most studies, we kept sections 0-1 as development set. In order to ensure we have a large enough test set to properly perform tests for statistical significance over F scores and balanced accuracies, we did not follow previous work (Lin et al., 2014; Park and Cardie, 2012) that used only section 23 or sections 23-24 for testing. Also, the traditional rule of thumb is to split the available data into training

[1]The rest of the data are EntRel/NoRel.

and testing sets with 80%/20% ratio. Our choice ensures that this is the case for all of the relations.

The only features that we use in our experiments are production rules. We exclude features that occur fewer than five times in the training set. Production rules are the state-of-the-art representation for discourse relation recognition. This representation leads to only slightly lower results than a system including a much larger variety of features in the first end-to-end PDTB style discourse parser (Lin et al., 2014) .

The production rule representation is based on the constituency parse of the arguments and includes both syntactic and lexical information. A production rule is the parent with an left-to-right ordered list of all of its children in the parse tree (for example, S→NP VP). All non-terminal nodes are included as a parent, from the sentence head to the part-of-speech of a terminal. Thus words that occur in each sentence augmented with their part of speech are part of the representation (for example, NN→company), along with more general structures of the sentence corresponding to production rules with only non-terminals on the right-hand side.

There are three features corresponding to a production rule, tracking if the rule occurs in the parse of first argument of the relation, in the second, or in both.

Adopting this representation allows us to focus on the issue of class imbalance and how the choices of tackling this problem affect eventual prediction performance. Our findings are representation-independent and will most likely extend to other representations.

We train and evaluate a binary classifier with linear kernel using SVMLight[2] (Joachims, 1999) for each of the four top level classes of relations: *Temporal*, *Comparison*, *Contingency* and *Expansion*. We used SVM-Multiclass[3] for standard multiway classification. We also develop and evaluate two approaches for multiway classification for the four classes plus the additional class of entity relation and no relation.

Due to the uneven distribution of classes, we use precision, recall and f-measure to measure binary prediction performance. For multiway classification, we use the balanced accuracy (BAC):

$$BAC = \frac{1}{k}\sum_{i=1}^{k}\frac{c_i}{n_i}, \quad (1)$$

where k is the number of relations to predict, c_i is the number of instances of relation i that are correctly predicted, n_i is the total number of instances of relation i.

Balanced accuracy (or averaged accuracy) has a more intuitive interpretation than F-measure. It is not dominated by the majority class as much as standard accuracy is. For example for two classes, in a dataset where one class makes up 90% of the data, predicting the majority class has accuracy of 90% but balanced accuracy of 45%.

In testing, we keep the original distribution intact and make predictions for all pairs of adjacent sentences in the same paragraph that do not have an explicit discourse relation [4]. In order to perform tests for statistical significance over F scores, precision, recall and balanced accuracies, we randomly partitioned the testing data into 10 groups. We kept the data distribution in each group as close as possible to the overall testing set. To compare the performance of two different systems, a paired t-test is performed over these 10 groups.

4 Why downsampling?

Binary classification As mentioned in the previous sections, in all prior work of supervised implicit relation classification, the technique to cope with highly skewed distribution for binary classification is to downsample the negative training instances so that the sizes of positive and negative classes are equal. The reason for doing so is that the classifier can achieve high accuracy just by ignoring the small class, learning nothing and aways predicting the larger class. We illustrate this effect in Table 2. Without downsampling, the only reasonable F measure is achieved for *Expansion* where the smaller class accounts for 40% of the data. Note that with downsampling, the recognition of *Expansion* is also improved considerably.

Multiway classification In prior work multiway classifiers are trained on all available training data. As we just saw, however, this approach leads

[2]http://svmlight.joachims.org/

[3]http://svmlight.joachims.org/svm_multiclass.html

[4]Note the contrast with prior work where in some cases *EntRel*s are part of *Expansion*, or in some cases the performance of methods is evaluated only on pairs of sentences where a discourse relation holds, excluding *EntRel*s, *NoRel*s or *AltLex*s.

	All data	Downsample
Temp.	0 (nan/0.0)	15.52 (8.8/65.4)
Comp.	2.17 (71.4/1.1)	27.65 (17.3/69.2)
Cont.	0.96 (100.0/0.5)	47.14 (34.5/74.5)
Exp.	44.27 (54.9/37.1)	55.42 (49.3/63.3)

Table 2: F measure (precision/recall) of binary classification: including all of the data vs downsampling.

to poor results in identifying the core *Temporal*, *Comparison* and *Contingency* discourse relations. We propose an alternative approach to multi-class prediction, based on binary one-against-all classifiers for each of the four discourse relations, including *Expansion*, trained using downsampling.

The intuition is that an instance of adjacent sentences S_i is assigned to a discourse relation R_j if the binary classifier for R_j recognizes S_i as a positive instance with confidence higher than that of the classifiers for other relations. If none of the binary classifiers recognizes the instance as a positive example, the instance is assigned to class *EntRel/NoRel*. This approach modifies the way multi-class classifiers are normally constructed by including downsampling and having special treatment of the *EntRel/NoRel* class.

Specifically, we first use the four binary classifiers C_j for each relation j to get the confidence p_j of instance i belonging to class j. We approximate the confidence by the distance to the hyperplane separating the two classes, which SVMLight provides. If at least one p_j is greater than zero, assign instance i the class k where the classifier confidence is the highest. If none of the p_j's is greater than zero, assign i to be the *EntRel/NoRel* class.

We show balanced accuracies of these two multiway classification methods in Table 3.

	Multiway SVM	One-Against-All
5-way	32.58	37.15

Table 3: Balanced accuracies for SVM-Multiclass and one-against-all 5-way classification.

The one-against-all approach leads to 5% absolute improvement in performance. A t-test analysis confirms that the difference is significant at $p < 0.05$. Note that the improvement comes entirely from acknowledging that skewed class distribution poses a problem for the task and by addressing the problem in the same way for binary and multi-class prediction.

5 Using more data

Although downsampling gives much better performance than simply including all of the original data, it still appears to be an undesirable solution because in essence it throws away much of the annotated data. This means that for the smallest relations, as much as 90% of the data will not be used. Feature selection and feature values are computed only based on this much smaller dataset and do not properly reflect the information about discourse relations encoded in the PDTB. In this section we first discuss some of the widely used methods for handling skewed data distribution, that is, weighted cost and upsampling. First, we show that with highly skewed distributions, the two methods result in almost identical classifiers. Then we introduce a method for feature selection and shaping which computes feature weights on the full dataset and thus captures much of the information lost in downsampling.

5.1 Weighted cost and upsampling

A number of methods have been developed for the skewed distribution problem (Morik et al., 1999; Veropoulos et al., 1999; Akbani et al., 2004; Batista et al., 2004; Chawla et al., 2002). Here we highlight weighted cost and random upsampling, which are known to work well and widely used.

The idea behind weighted cost (Morik et al., 1999; Veropoulos et al., 1999) is to use weights to adjust the penalties for false positives and false negatives in the objective function. As in Morik et al. (1999), we specify the cost factor to be the ratio of the size of the negative class vs. that of the positive class.

In the case of upsampling, instead of randomly downsampling negative instances, positive instances are randomly upsampled. In our experiments we randomly replicate positive instances with replacement until the numbers of positive and negative instances are equal to each other.

The binary and multiway classification results for these two methods are shown in Table 4 and Table 5. For binary classification, we can see significantly higher F score for the smallest *Temporal* class. Weighted cost is also able to achieve significantly better F-score for *Expansion*. For *Comparison* and *Contingency*, the F-scores are similar to that of plain downsampling. The balanced accuracies of multi-class classification with either methods are lower, or significantly lower in the case of

weighted cost, than using downsampling in one-against-all manner.

	Upsample	WeightCost
Temp.	20.35* (16.8/25.9)	20.61* (16.9/26.3)
Comp.	28.11 (20.6/44.5)	28.38 (19.9/49.6)
Cont.	46.46 (37.4/61.3)	46.36 (34.6/70.1)
Exp.	54.93 (50.3/60.5)	57.43* (43.9/83.1)

Table 4: F-measure (precision/recall) of binary classification: upsampling vs. weighted cost.

For *Temporal* and *Comparison* relations listed in Table 4, we noticed an interesting similarity between the F and precision values of upsampling and weighted cost. To quantify this similarity, we calculated the Q-statistic (Kuncheva and Whitaker, 2003) between the two classifiers. The Q-statistic is a measurement of classifier agreement raging between -1 and 1, defined as:

$$Q_{w,u} = \frac{N_{11}N_{00} - N_{01}N_{10}}{N_{11}N_{00} + N_{01}N_{10}} \quad (2)$$

Where w denotes the system using weighted cost, u denotes the upsampling system. N_{11} means both systems make a correct prediction, N_{00} means both systems are incorrect, N_{10} means w is incorrect but u is correct, and N_{01} means w is correct but u is incorrect.

We have the following Q statistics: *Temporal*: 0.999, *Comparison*: 0.9938, *Contingency*: 0.9746, *Expansion*: 0.7762. These are good indicators that for highly skewed relations, the two methods give classifiers that behave almost identically on the test data. In the discussions that follow, we discuss only weighted cost to avoid redundancy.

5.2 Feature selection and shaping

While weighted cost or upsampling can give better performance over downsampling for some relations, their disadvantages towards multi-class classification and the obvious favor towards the majority class give rise to the following question: is it possible to inform the classifier of the information encoded in the annotation of *all* of the data while still using downsampling to handle the skewed class distribution? Our proposal is feature value augmentation. Here we introduce a relational matrix in which we calculate augmented feature values via feature shaping. We first compute the values of features on the entire training set, then use the downsampled set for training with these values. In this way we pass on to the classifiers information about the relative importance of features gleaned from the entire training data.

5.2.1 Feature shaping

The idea of feature shaping was introduced in the context of improving the performance of linear SVMs (Forman et al., 2009). In linear SVMs the prediction is based on a linear combination of *weight*×*feature values*. The sign of *weight* indicates the preference for a class (positive or negative), the value of the feature should correspond to how strongly it indicates that class. Thus, features that are strongly discriminative should have high values so that they can contribute more to the final class decision. Here we augment feature values for a relation according to the following criteria: 1. Features are considered "good" if they strongly indicate the *presence* of the relation; 2. Features are considered "good" if they strongly indicate the *absence* of the relation; 3. features are considered "bad" if their presence give no information about *either the presence or the absence* of the relation.

To capture this information, we first construct a relation matrix M with each entry M_{ij} defined as the conditional probability of relation R_j given the feature F_i computed as the maximum likelihood estimate from the full training set:

$$M_{ij} = P(R_j|F_i)$$

Each column of the relation matrix captures the predictive power of each feature to a certain relation. A feature with value M_{ij} higher than the column mean indicates that it is predictive for the presence of relation j, while a feature with M_{ij} lower than the mean is predictive for its absence; the strength of such indication depends on how far away M_{ij} is from the mean: the further away it is, the more valuable this feature should be for relation j. With this idea we give the following augmented value for each feature:

$$M'_{ij} = \begin{cases} M_{ij}, & \text{if } M_{ij} \geq \mu_j. \\ \mu_j + (\mu_j - M_{ij}), & \text{if } M_{ij} < \mu_j. \end{cases} \quad (3)$$

where μ_j is the mean of the jth column corresponding to the jth relation.

Given a feature F_i, very small and very high probabilities of a certain relation j, i.e., $P(R_j|F_i)$, are both useful information. However, in linear SVMs, lower values of a feature would mean that it contributes less to the decision of the class. By

feature shaping, we allow features that strongly indicate the absence of a class to influence the decision and rely on the classifier to identify the negative association and reflect it by assigning a negative weight to these features.

When constructing the relation matrix, we used the top four relation classes along with an *EntRel/NoRel* class. We computed the matrix before downsampling to preserve the natural data distribution and features that strongly indicate the absence of a class, then downsample the negative data just like the previous downsampling setting.

5.2.2 Feature selection

The relation matrix also provides information for feature selection using a binomial test for significance, $B(n, N, p)$, which gives the probability of observing a feature n times in N instances of a relation if the probability of any feature occurring with the relation is p. For each relation, we use the binomial test to pick the features that occur significantly more or less often than expected with the relation. In the binomial test, p is set to be equal to the probability of that relation in the PDTB training set. We select only the features which result in a low p-value for the binomial test for at least some relation. We used 9-fold cross validation on the training data to pick the best p-values for each relation individually; all best p-values were between 0.1 and 0.2.

Result listing Table 5 and Table 6 show the multiway and binary classification performance using feature shaping and feature selection. We also show the precision and recall for binary classifiers.

	Multiway SVM	One-Against-All
AllData	32.58	NA
Downsample	NA	37.15
Upsample	NA	36.63
Weighted Cost	NA	34.23
Selection	32.52	38.42*
Shaping	NA	38.81**
Shape+Sel	NA	39.13**

Table 5: Balanced accuracy for multiway SVM and one-against-all for 5-way classification. One asterisk (*) means significantly better than weighted cost and upsampling, and two means significantly better than downsampling, at $p < 0.05$.

For multi-way classification, performing feature shaping leads to significant improvements over downsampling, upsampling and weighted cost. The binomial method for feature selection that relies on the full training data distribution has a similar effect. Combined feature shaping and selection leads to 2% absolute improvement in discourse relation recognition. For binary classification, though, the improvement is significant only for *Temporal*.

6 Classifier analysis and combination

6.1 Discussion of precision and recall

A careful examination of Tables 5 and 6 leads to some intriguing observations. For the most skewed relations, if we consider not only the F measure, but also the precision and recall, there is an interesting difference between the systems. While downsampling has the lowest precision, it gives the highest recall. The case for weighted cost is another story. For highly skewed relations such as *Temporal* and *Comparison*, it gives the highest precision and the lowest recall; but as the data set balances out in downsampling, the classifier shifts towards high recall and low precision.

We can also rank the three feature augmentation techniques in terms of how much they reflect distributional information in the training data. Feature selection reflects the training data least among the three, because it uses information from all of the data to select the features, but the feature values are still either 1 or 0. Feature shaping engages more data because the value of a feature encodes its relative "effectiveness" for a relation. We can see that feature selection gives slightly higher precision than just downsampling; feature shaping, on the other hand, gives precision and recall values between these two. This is most obvious in smaller relations, i.e. *Temporal* and *Comparison*.

To see if this trend is statistically significant, we did a paired t-test over the precision and recall for each system and each relation. For the *Temporal* relation, all systems that use more data have significantly higher ($p < 0.05$) precision than that for downsampling. For *Comparison*, the changes in precision are either significant or tend towards significance for three methods: feature shaping ($p < 0.1$), feature shaping+election ($p < 0.1$) and weighted cost ($p < 0.05$). For *Contingency*, feature shaping gives an improvement in precision that tends toward significance ($p < 0.1$). The drops in recall using feature shaping or weighted cost for the above three relations are significant ($p < 0.05$). For the *Expansion* relation, being the largest class with 40% positive data, changes in

	Downsample	WeightCost	Selection	Shaping	Shape+Sel
Temp.	15.52 (8.8/65.4)	20.61* (16.9/26.3)	18.47* (10.7/65.9)	20.37* (12.6/53.2)	21.30* (13.7/47.8)
Comp.	27.65 (17.3/69.2)	28.38 (19.9/49.6)	26.98 (17.4/60.1)	27.79 (18.3/58.2)	26.92 (18.7/48.2)
Cont.	47.14 (34.5/74.5)	46.36 (34.6/70.1)	47.45 (34.7/75.2)	47.62 (35.4/72.9)	46.93 (35.2/70.5)
Exp.	55.42 (49.3/63.3)	57.43* (43.9/83.1)	55.52 (49.3/63.5)	55.13 (49.3/62.5)	54.90 (49.2/62.1)

Table 6: F score (precision/recall) of classifiers with feature augmentation. Asterisk(*) means F score or BAC is significantly greater than plain downsampling at $p < 0.05$.

precision and recall with downsampling systems are not significant; yet weighted cost shifted towards predicting more of the positive instances, i.e., giving a significantly higher recall by trading with a significantly lower precision ($p < 0.05$).

6.2 Discussion of classifier similarity

To better understand the differences of classifier behaviors under the weighted cost and each downsampling technique (plain downsampling, feature selection, feature shaping, feature shaping+selection), in Table 7 we show the percentage of test instances that the weighted cost system and each downsample system agree or do not agree. In particular, we study the following situations:

1. The downsample system predicts correctly but the weighted cost system does not ("D+C-");

2. The weighted cost system predicts correctly but the downsample system does not ("D-C+");

3. Both systems are correct ("D+C+").

At a glance of the Q statistic, it seems that the systems are not behaving very differently. However, as only the sum of disagreements is reflected in the Q statistic, we look more closely at where the systems do not agree in each situation. If we focus on the rarer *Temporal* and *Comparison* relations, first note that in the plain downsampling vs. weighted cost, the percentage of test instances in the "D+C-" column is much smaller than that in the "D-C+" column. This aligns with the above observation that plain downsampling gives much lower precision for these relations than weighted cost. Now, as more data is engaged from first using feature selection, then using feature shaping, then using both, the percentage of instances where both systems predict correctly increase. At the same time, there is a drop in the percentage of test instances in the "D-C+" column. This trend is also a reflection of the observation that as more data is engaged, the precision got higher as the recall drops lower. As the data gets more evenly distributed, this phenomenon fades away. The table also reveals a subtle difference between feature shaping and feature selection. Compared to downsampling, feature selection introduces an increase in the column "D+C-" (i.e. the weighted cost system makes a mistake but the downsample system is correct). Feature shaping, on the other hand, do not necessarily increase this new kind of difference between classifiers.

	D+C- (%)	D-C+ (%)	D+C+ (%)	Q Stat
Temporal				
Downsamp	2.56	28.27	61.47	0.73
Selection	2.91	22.04	67.71	0.77
Shaping	2.61	13.36	76.39	0.89
Sel+Shape	2.83	10.42	79.32	0.90
Comparison				
Downsamp	5.74	18.24	53.76	0.84
Selection	7.72	16.14	55.85	0.80
Shaping	6.14	11.95	60.04	0.89
Sel+Shape	9.69	10.99	61.01	0.83
Contingency				
Downsamp	6.88	7.89	58.74	0.93
Selection	8.01	8.92	57.70	0.91
Shaping	7.07	6.73	59.90	0.94
Sel+Shape	8.68	8.13	58.49	0.91
Expansion				
Downsamp	16.39	8.23	44.66	0.82
Selection	17.87	9.71	43.18	0.76
Shaping	16.64	8.45	44.44	0.81
Sel+Shape	18.36	10.30	42.59	0.73

Table 7: Q statistics and agreements (in percentages) of each downsampling system vs. weighted cost. "D" denotes the respective downsample system in the left most column; "C" denotes the weighted cost system. A "+" means that a system makes a correct prediction; a "-" means a system makes an incorrect prediction.

6.3 Classifier combination

Our classifier comparisons revealed that for highly skewed distributions, there are consistent differences in the performance of classifiers obtained by using the training data in different ways. It stands to reason that a combination of these classifiers with different strengths will result in an overall improved classifier. This idea is explored here.

Suppose on a sample i, the downsampling classifier predicts the target class with confidence p_{id}, and the weighted cost classifier predicts the target

class with confidence p_{ic}. Here again we approximate the confidence of the class by the distance from the hyperplane dividing the two classes. We weight the two predictions and get a new prediction confidence by:

$$p'_i = \frac{\alpha_d p_{id} + \alpha_u p_{ic}}{\alpha_d + \alpha_c}. \quad (4)$$

where the αs are parameters we want to encode how much we trust each classifier. To get these values, we train the classifiers and get the accuracies from each of them on the development set. Since we are using linear SVMs in our experiments, we mark the sample as positive if $p_i > 0$, and negative otherwise.

The results for the combination are shown in Table 8. We include the original performances of the classifiers by themselves for reference.

F measure For *Temporal*, the combined classifier performs better than the original classifiers. We see significant ($p < 0.05$) improvements over the corresponding downsampling system and the weighted cost system. If feature shaping is involved in the combination, it is also having better performance that tend toward significance ($p < 0.1$) over the weighted cost classifier. For *Comparison*, the benefits of a combined system is also obvious for feature shaping and/or selection. Feature shaping combined with weighted cost gives significantly ($p < 0.05$) better performance than either of them individually, and feature selection and shaping+selection combined with weighted cost is better than themselves alone. For *Contingency*, though weighted cost do not give better results, the improvement tends toward significance ($p < 0.1$) when combined with plain downsampling. For *Expansion* where weighted cost gives the lowest precision, combination with other classifiers do not give significant improvements over F scores.

Precision and recall We can also compare the precision and recall for each system before and after combination. In all but one cases for *Temporal* and *Comparison*, we observe significantly higher precision and much lower recall after the combination. The case for *Expansion* is just the opposite as expected.

7 Conclusion

In this paper, we studied the effect of the use of annotated data for binary and multiway classification in supervised implicit discourse relation recognition. The starting point of our work was to establish the effectiveness of downsampling negative examples, which was practiced but not experimentally investigated in prior work. We also evaluated alternative solutions to the skewed data problem, as downsampling throws away most of the data. We examined the effect of upsampling and weighted cost. In addition, we introduced the relation matrix to give more emphasis on informative features through augmenting the feature value via feature shaping. We found that as we summarize more detailed information about the data in the full training set, performance for multiway classification gets better. We also observed through precision and recall that there are fundamental differences between downsampling and weighted cost, and this difference can be beneficially exploited by combining the two classifiers. We showed that our way of doing such combination gives significantly higher performance results for binary classification in the case of rarer relations.

	Original Classifier	Combined Classifier
Temporal		
WeightCost	20.61 (16.9/26.3)	
Downsamp	15.52 (8.8/65.4)	21.78* (14.9/40.5)
Selection	18.47 (10.7/65.9)	22.99* (15.8/42.0)
Shaping	20.37 (12.6/53.2)	23.88* (17.5/37.6)
Sel+Shape	21.30 (13.7/47.8)	23.72* (17.7/36.1)
Comparison		
WeightCost	28.38 (19.9/49.6)	
Downsamp	27.65 (17.3/69.2)	28.72 (19.3/56.4)
Selection	26.98 (17.4/60.1)	29.25* (20.1/54.0)
Shaping	27.79 (18.3/58.2)	29.89*$^+$ (20.5/54.9)
Sel+Shape	26.92 (18.7/48.2)	29.83* (21.3/50.0)
Contingency		
WeightCost	46.36 (34.6/70.1)	
Downsamp	47.14 (34.5/74.5)	48.38$^+$ (35.9/74.4)
Selection	47.45 (34.7/75.2)	47.76$^+$ (35.5/72.9)
Shaping	47.62 (35.4/72.9)	48.16$^+$ (36.0/72.9)
Sel+Shape	46.93 (35.2/70.5)	47.37 (35.6/70.7)
Expansion		
WeightCost	57.43 (43.9/83.1)	
Downsamp	55.42 (49.3/63.3)	56.61* (46.4/72.7)
Selection	55.52 (49.3/63.5)	57.10* (46.5/73.0)
Shaping	55.13 (49.3/62.5)	56.74* (46.4/73.0)
Sel+Shape	54.90 (49.2/62.1)	57.06* (46.4/74.0)

Table 8: Classifier combination results for binary classification. An asterisk(*) means significantly better than the corresponding downsampling system at, and a plus(+) means significantly better than weighted cost, at $p < 0.05$. Improvements that tend toward significance ($p < 0.1$) are not shown here but are discussed in the text.

References

Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. 2004. Applying support vector machines to imbalanced datasets. In *Machine Learning: ECML 2004*, pages 39–50.

Gustavo E. A. P. A. Batista, Ronaldo C. Prati, and Maria Carolina Monard. 2004. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter - Special issue on learning from imbalanced datasets*, 6(1):20–29, June.

Or Biran and Kathleen McKeown. 2013. Aggregated word pair features for implicit discourse relation disambiguation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL): Short Papers*, pages 69–73.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, June.

R. Elwell and J. Baldridge. 2008. Discourse connective argument identification with connective specific rankers. In *IEEE International Conference on Semantic Computing (IEEE-ICSC)*, pages 198 –205.

George Forman, Martin Scholz, and Shyamsundar Rajaram. 2009. Feature shaping for linear SVM classifiers. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 299–308.

Sucheta Ghosh, Richard Johansson, Giuseppe Riccardi, and Sara Tonelli. 2011. Shallow discourse parsing with conditional random fields. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 1071–1079.

Barbara J. Grosz, Scott Weinstein, and Aravind K. Joshi. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21:203–225.

Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. 2010. A semi-supervised approach to improve classification of infrequent discourse relations using feature vector extension. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 399–409.

Yu Hong, Xiaopei Zhou, Tingting Che, Jianmin Yao, Qiaoming Zhu, and Guodong Zhou. 2012. Cross-argument inference for implicit discourse relation recognition. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*, pages 295–304.

Thorsten Joachims. 1999. Making large-scale support vector machine learning practical. In *Advances in kernel methods*, pages 169–184.

Ludmila I. Kuncheva and Christopher J. Whitaker. 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, May.

Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. 2009. Recognizing implicit discourse relations in the Penn Discourse Treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 343–351.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2014. A PDTB-styled end-to-end discourse parser. *Natural Language Engineering*, 20:151–184, 4.

Eleni Miltsakaki, Livio Robaldo, Alan Lee, and Aravind Joshi. 2008. Sense annotation in the Penn Discourse Treebank. In *Proceedings of the 9th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, pages 275–286.

Katharina Morik, Peter Brockhausen, and Thorsten Joachims. 1999. Combining statistical learning with a knowledge-based approach - a case study in intensive care monitoring. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML)*, pages 268–277.

Joonsuk Park and Claire Cardie. 2012. Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 108–112.

Emily Pitler, Mridhula Raghupathy, Hena Mehta, Ani Nenkova, Alan Lee, and Aravind Joshi. 2008. Easily identifiable discourse relations. In *Proceedings of the International Conference on Computational Linguistics (COLING): Companion volume: Posters*, pages 87–90.

Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL-IJCNLP)*, pages 683–691.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*.

Konstantinos Veropoulos, Colin Campbell, and Nello Cristianini. 1999. Controlling the sensitivity of support vector machines. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, volume 1999, pages 55–60.