Identifying Justifications in Written Dialogs

Or Biran
Department of Computer Science
Columbia University
orb@cs.columbia.edu

Owen Rambow
CCLS
Columbia University
rambow@ccls.columbia.edu

***Abstract*—In written dialog, discourse participants need to justify claims they make, to convince the reader the claim is true and/or relevant to the discourse. This paper presents a new task (with an associated corpus), namely detecting such justifications. We investigate the nature of such justifications, and observe that the justifications themselves often contain discourse structure. We therefore develop a method to detect the existence of certain types of discourse relations, which helps us classify whether a segment is a justification or not. Our task is novel, and our work is novel in that it uses a large set of connectives (which we call indicators), and in that it uses a large set of discourse relations, without choosing among them.**

I. Introduction

Natural language processing has recently seen a proliferation of interest in genres other than newswire. In particular, written dialog such as email and web-based discussion forums and blogs have been attracting a lot of attention, as they show a lot of interesting uses of language which newswire and other highly monologic and purely informative genres do not. For example, consider subjective statements and attempts to justify such statements. While newswire may contain editorials, these are the exception. In contrast, in spontaneously user-generated online content, we find many subjective statements, and, partly because of the interactive nature of the medium, many attempts at justifying such statements.

In this paper, we address the problem of identifying justifications for subjective claims in interactive written dialogs.[1] We do not take “justification” to mean a particular discourse relation (such as the JUSTIFY relation from RST). Instead, we use this term in a broad dialogic sense: the writer makes an utterance which conveys subjective information (in the sense of [1]) and anticipates the question “Why are you telling me that?”. Put differently, she is showing the reader that she is being relevant in a Gricean sense [2], presumably in an attempt to engage the reader and have him continue reading.

Here are some examples of what we consider to be justification, taken from our corpus of discussions following blog posts (the provided categories are only for explanatory purposes, they are not part of the task we address):

(1) Recommendation for action, and motivation for proposed action:
Claim: I’d post an update with the new date immediately
Justification: in case anyone makes plans between now and when you post the reminder.

(2) Statement of like or dislike or of desires and longing, and subjective reason for this like or dislike or desire or longing:
Claim: This is a great, great record.
Justification: I’m hesitant to say that kind of thing because I’m not a critic; but it is certainly in a league with Robyn’s very best work. The Venus 3 come together as a band in a way I don’t think they really did on O’ Tarantula, and it just touches me very deeply.

(3) Statement of like or dislike or of desires and longing, and claimed objective reason for this like or dislike or desire or longing:
Claim: Song of the South should be released again.
Justification: It is not racist. Uncle Remus was a slave and the stories came from slavery days. While slavery was a horrible thing, we cant just act like it never happened.

(4) Statement of subjectively perceived fact, with a proposed objective explanation:
Claim: I don’t think Wilf will die.
Justification: Wilf’s going to have to kill Ten to save Donna or something, ’cause of the whole ’you’ve never killed a man’ thing that TV woman said.

(5) A claimed general objective statement and a more specific objective statement that justifies the more general one.
Claim: But it always leads to lives and potential unfulfilled when love is thwarted...and depression, too.
Justification: We see a hell of a lot of that still, judging by the number of gay and bi men who are on anti-depressants and in therapy

What is striking is that all but one (1) of these justifications are not atomic discourse units, but contain argumentation themselves: in order to justify a claim, the writer is presenting an entire argument. For example, in (5), the justification contains two parts: an empirical claim, and then evidence for that claim. The reader, however, interprets the entire passage as the justification of the original claim. Thus, we are interested in detecting argumentation in support of a given claim, such that the entire argumentation is considered the justification for the claim by the reader. As a consequence, in this paper, we are not

[1] This effort is part of a larger effort to detect attempts to persuade in written dialog; however, we believe that the research presented in this paper is of interest beyond the motivating application, for example, for categorizing text or passages as expositive *versus* argumentative.

interested in detecting a single discourse relation, for example one of those proposed by Rhetorical Structure Theory (RST) [3] or the theory underlying the Penn Discourse Treebank [4]. Instead, we are interested in a type of discourse contribution, which frequently is characterized as containing argumentation. However, this argumentation is also not characterized by a single discourse relation; instead, it can be realized by a large number of discourse relations. As a consequence, recent work on identifying discourse relations [5]–[7] is only relevant as a building block, but it is not the solution to our problem. Instead, we use a multi-step approach:

- We extract lists of indicators for a number of relations from the RST Treebank, a news corpus annotated for RST relations.
- We extract a list of co-occurring content word pairs for each of the indicators from a large, multi-topic corpus (English Wikipedia).
- We use the lists of pairs to formulate features for a machine learning model and apply it to the task of identifying justifications in a corpus of online discussion.

Crucially, we do not apply these new features to both the claim and the candidate justification: we only look at the candidate justification, with the assumption that the justification frequently includes complex discourse relations. In experiments looking at both claim and candidate (which are not described in this paper) we observed that including the claim consistently adds nothing or very little (less than 0.5%) to all of our systems.

While our work is in the context of a larger project which aims at identifying both the claim and its justification, in this paper we only report on the automatic detection of justifications, given gold-standard claims. The identification of potential claims is related to the identification of subjectivity, and thus falls in a completely different line of research.

This paper is structured as follows. In Section II, we provide an overview of related work. We then introduce our data and our annotation in Section III. We start out by presenting fully supervised learning experiments on this corpus; these experiments provide a baseline for this paper (Section IV). We then investigate the use of additional features obtained through unsupervised methods (Section V). We finish with a discussion (Section VI) and a conclusion (Section VII).

II. Related Work

As we explained in the introduction, our work is novel and different from other work in that we are not interested in finding discourse relations from a specific pre-defined set. In this section, we briefly review previous research that attempts to identify specific discourse relations, as we draw on the techniques developed by those researchers.

The principal idea here is the line of research started by [8], who use unsupervised methods to increase the recognition of implicit relations, i.e., relations not signaled by a cue word. The basic technique is simple: sentences with explicit connectives are used to train models to recognize cases of implicit relations; the underlying assumption is that implicit relations and explicitly signaled relations do not differ greatly in terms of the content of the related segments. These techniques were further developed by [5], [7]. Critical assessments of this approach can be found in [9] and [6]. [6] do an extensive study using the Penn Discourse Treebank [4], and observe that many of the meaningful word pairs learned from unannotated data involve closed-class words (which contradicts the intuition that these word pairs represent semantic relations), and that the models derived from relations with explicit connectives do not, in fact, work very well on relations with implicit connectives.

III. Data

We use three corpora in the different stages of our system.

The RST Treebank [10] is a subset of the Wall Street Journal part of the Penn Treebank, annotated with discourse relations based on Rhetorical Structure Theory (RST). We use the RST Treebank to extract relation indicators. We do not use the Penn Discourse Treebank directly, though we could have used it instead of the RST Treebank.

For our unsupervised word pair extraction, we use English Wikipedia[2]. We pre-processed the corpus to remove HTML tags, comments, links and text included in tables and surrounding figures (e.g, captions and descriptions). The remaining text was lowercased and split into sentences.

Finally, we run our system on a corpus of 309 blog threads from LiveJournal[3], belonging to users from English-speaking countries. Each thread contains an original entry by the blog owner and a set of comment entries, in a tree structure[4], by other LJ users as well as the owner. The threads contain annotated *claims* and their corresponding *justifications*. A justification can only be made by the same poster who made the original claim, but it may be located in a different entry. All annotated claims have justifications, and a claim may have more than one justification. 32.4% of the justifications are in the sentence following the claim; 97.3% are in the same entry as the claim; 77.6% appear after the claim.

In inter-annotator agreement calculations on a subset of the data (including only those claims which were marked by both annotators, with candidate justification sentences chosen in the same way they are chosen during a run of the system) we observed a kappa measure of 0.69 and an f-measure between annotators of 0.75.

One important attribute of the LiveJournal corpus is that there is wide variation among the threads: the standard deviations of the entry length, both in words and in sentences, are higher than the mean; that is also the case for claims per thread and claims per entry.

IV. Sentence Pair Classification: Fully Supervised Learning

The task is deciding for a pair of sentences, the first of which is marked the *claim*, whether or not the second sentence is a justification of the claim.

[2] A snapshot of all article texts as of April 8th, 2010

[3] http://www.livejournal.com

[4] That is, a comment is associated with a particular previous entry, which can itself be a comment or the original post.

Our sentence pairs come from LiveJournal blog threads. We describe the exact way in which we create our data set of pairs in a later section. Here we describe the classification system we used as a baseline.

A. Naive Baseline

To put things in context, we provide the results of a very naive baseline which simply chooses the sentence immediately following each claim as its justification.

B. Heuristic Baseline

We achieve a better baseline performance for our task using a heuristic system with the following rules:

1) If the claim is not in the same entry as the candidate justification, classify as NO.
2) If the distance, in number of sentences, between the claim and the candidate justification is more than 4, classify as NO.
3) Otherwise classify as YES.

This very basic system achieves some performance, and in particular has high recall ($> 91\%$ in cross-validation - See Table II).

C. Hybrid Baseline

Because the heuristic system achieves some precision with very little sacrifice of recall, we use it as a first stage in all systems in our experiments. These hybrid systems first pass the data through the first two rules above; if a data point passes, then it is sent to a supervised learning classifier. Our hybrid baseline classifier operates on only two simple features: *beforeClaim*, a binary feature signifying whether the justification candidate comes before or after the claim, and *sentenceLength*, the length (in words) of the justification candidate. We found that justifications are longer on average than other sentences. We tried to match the claim to the justification in ways other than distance, by adding word overlap features with various variations including the use of stemming, n-grams, and WordNet for synonymy resolution. However, none of these attempts increased performance for this task.

While claims are allowed to have multiple justifications, it is rare that they have more than a few. To avoid picking too many sentences as justifications for a single claim we added a post-processing stage that looks at the pairs which share a claim and prevents all but the two with the highest confidence from being classified as justifications. Two was found the be the optimal number in a manual tuning.

This is the real baseline used in our experiment; the lesser two baseline results are shown for completeness. In all systems of Table II which are described as "baseline + X", the baseline is the hybrid baseline containing these two features with post-processing.

D. Bag of Words Baseline

Finally, we include a full system (the hybrid baseline plus additional features) as a baseline. The additional features in this case are the standard Bag of Words features: we used all non-punctuation tokens which appear more than 5 times in the data set, each as a separate feature, for a total of 1474.

V. Adding Features Obtained Through Unsupervised Mining

Particular RST relations, such as cause, concession or contrast may indicate argumentation.

Discovering RST relations in text is not a simple task. Some relations typically contain a connector word or phrase - such as *but* for the contrast relation - but sometimes it may be implicit or replaced with a paraphrase (for example, *but* may be replaced with *on the other hand*). In online dialog especially, we expect more frequent irregularities in the usage of standard connectors. In addition, many such connectors are not reliable indicators even when present, since they tend to be common, ambiguous words. Still other relations make rare or no use of connectors at all.

The idea driving our method is that some word combinations are more likely to appear as part of a relation. A simple example for contrast are antonyms - for instance, *easy* and *difficult* in the following sentence from LiveJournal:

> Its easy to flatter people, but its difficult to tell the truth and say something honest that might sound mean.

More generally, words may have a likely causal or even more subtle relationship between them. Consider the causality between *fresh* and *best* in:

> Rum tastes best when it's still relatively fresh and you can still taste the cane.

The concession indicated by *horrible* and *happened* in:

> While slavery was a horrible thing, we cant just act like it never happened.

Or the elaboration evidenced by *photography* and *sensor* (as well as other possible pairs) in:

> Canon provide an overall better photography system, from body to sensor to optics (canon Lseries lenses are something out of this world).

Crucially, the word pairs above are *content words*, which are independent of the linguistic style and even grammaticality of the text in question. We should expect such pairs to be relevant to a variety of corpora, with the reservation that domain may have much to do with the frequency of their appearance. We chose Wikipedia as the corpus from which to extract pairs in order to minimize the dominance of domain-specific pairs, since Wikipedia articles deal with a variety of topics.

A. Extracting Indicators

We extracted a list of indicators from the RST Treebank. Unlike the PDTB, which has a list of indicators that are used (explicitly or implicitly) for each relation, the RST Treebank simply specifies that two or more spans of text have

TABLE I
THE RELATIONS FOR WHICH INDICATORS WERE EXTRACTED, WITH THE NUMBER OF INDICATORS AND A FEW SAMPLES.

Relation	Nb	Sample indicators
analogy	15	as a, just as, comes from the same
antithesis	18	although, even while, on the other hand
cause	14	because, as a result, which in turn
concession	19	despite, regardless of, even if
consequence	15	because, largely because of, as a result of
contrast	8	but the, on the other hand, but it is the
evidence	7	attests, this year, according to
example	9	including, for instance, among the
explanation-argumentative	7	because, in addition, to comment on the
purpose	30	trying to, in order to, so as to see
reason	13	because, because it is, to find a way
result	23	resulting, because of, as a result of

a particular relation between them. We aim to automatically create a list of the most relevant n-grams for each relation, and choose our indicators from among the top candidates.

Specifically, our method works as follows.

We first choose n relations which we view as relevant for our task. We chose relations which relate to increasing the reader's willingness to accept a claim. RST [3] distinguishes presentational relations from subject-matter relations; the former are defined in terms of changes in the reader's strength of belief, desire, or intention, while the latter are defined in terms of making the reader entertain a new proposition, such as causality. Basically, we are interested in presentational relations. However, as [11] point out, subject matter relations can co-exist with presentational relations: claiming a causal relation between two events may well be the best way of convincing the reader that the caused (or the causing) event did indeed happen. Thus, we choose among both presentational and subject-matter relations those which are most likely to be usable in an attempt to make the reader accept a previously made claim.

For our experiments, we originally chose 14 relations. The RST Treebank uses a superset of a subset of the original relations proposed by RST . Specifically, MOTIVATION and JUSTIFY are not in the RST Treebank – we would have used them if they were. We excluded mainly subject-matter relations, specifically relations which are purely semantic such as MANNER, MEANS, or TEMPORAL-SAME-TIME; topic- and structure-related relations such as LIST, SUMMARY or TOPIC-SHIFT; and BACKGROUND, the only presentational relation we excluded, since its effect is to increases the reader's *ability* to understand the presented material, not necessarily his or her inclination to do so. During experiments, we discarded two of these, ATTRIBUTION and RHETORICAL-QUESTION, since they had no effect on the results, and were left with the 12 relations shown in Table I.

After choosing our relations, we create a set of documents $D = \{d_1..d_n\}$, where each document d_i contains all the text from the RST Treebank participating in relation i. The two spans of text participating in a relation (identified as such by the corpus) are retained as a single line.

We compute the top ngrams with a variant of tf-idf. We do the following for unigrams, bigrams, trigrams and 4-grams:

1) Extract all n-grams from all documents
2) Compute idf for each n-gram in the usual way
3) Compute for each n-gram j in each document d_i the tf variant tf$*_{ij} = \frac{l_{ij}}{\sum_k l_{ik}}$ where l_{ik} is the number of lines in d_i in which the n-gram k appears at least once. The intuition for this altered measure is that since each line corresponds to one instance of the relation, an n-gram appearing multiple times in the same line would be overweighted with the standard measure
4) Create a list of n-grams for each document sorted by tf*-idf

We delete all n-grams below a certain tf*-idf score. We used 0.004 as the cutoff value in all experiments. Some filtering was needed as it was not feasible to go over the entire lists (in the next stage below), and this was casually observed as a reasonable cutoff.

Finally, we manually went over the lists and deleted n-grams that seemed irrelevant, ambiguous or domain-specific. Many n-grams that appear even at the very top for some relations are clearly not relevant, mostly because of the relatively narrow domain of the RST Treebank. For example, the highest-ranking trigram for the EVIDENCE relation is *as a result*, which is reasonable; the next down the list, however, is *in New York* - clearly a product of the particular corpus. This manual culling only took place once, and the resulting list is publicly available[5]. It can be used to extract pairs from any corpus in an unsupervised way, as explained in Section V-B.

At the end of this process, we are left with 69 indicators in total, some of which are shared between multiple relations. Table I shows the number of indicators and a few samples for each relation.

B. Extracting WordPairs

Having finalized the list of indicators, we use it to extract word pairs from English Wikipedia. We split the corpus into sentences, remove sentences longer than 50 words in length[6], and for each indicator in our list, extract a list of word pairs

[5] at http://www.cs.columbia.edu/~orb

[6] Sentences longer than 50 words constitute only 2.7% of all Wikipedia sentences. Longer sentences are likely to be syntactically complex and thus too noisy for this method.

occuring in sentences at which the indicator occurs. We extract two lists of word pairs:

- The *sides* list, where the first word must occur to the left of the connector and the second must occur to the right of the connector. This set contains 447,149,688 pairs.
- The *anywhere* list, where the words may occur anywhere in the sentence but the first word must occur earlier. This set contains 1,017,190,824 pairs.

The words participating in the indicator itself are not considered for either of the lists. Stoplisted words (using the list of [12]) are also not considered. When stop words are allowed to participate in the pairs performance decreases: we include the results for a system which uses a set of pairs extracted without a stoplist in Table II, for comparison with our better-performing systems. Interestingly, [6] report the opposite – that removing stop words hurts their performance. An explanation for this, owing to the nature of our features, is given in the next section.

We collect frequency information - that is, how many times each word pair appears in the corpus. Pairs which appear less than 20 times are removed from the lists to reduce noise, but the frequency of the remaining pairs is not used in subsequent steps. After this filtering, the size of the *sides* list is 334,925 pairs and the size of the *anywhere* list is 719,439 pairs.

Although this method misses cases where the indicator is implicit and, for the *sides* list, cases where the indicator is sentence-initial, the abundance of data still allows the collection of large sets of word pairs.

C. Using the Information

In our experiments, we used a supervised classifier to decide whether or not in a pair of given sentences, the first of which is given as the *claim*, the second is a justification of the first. We describe the classifier and preparation in more detail in the next section; this section describes the various ways in which we formulated machine learning features from the data (indicators and word pairs) extracted in the previous sections.

We tried several approaches for using the extracted indicators and pairs:

1) Indicators as Lexical Features: In this simple approach, we used the indicators themselves as binary lexical features. The results were not positive, perhaps because phrases which are good indicators are too rare in the data while common phrases are not very good indicators.

2) Word Pair Features: Here we used the extracted pairs from the two sets, *sides* and *anywhere*, to build features. In order to avoid a sparse feature space we took advantage of the natural structure of the pair lists - namely, the fact that each pair is associated with one or more indicators. We created 69 features, one for each indicator, where each feature ϕ_j is associated with a set of pairs P_j, and

$$\phi_j = \begin{cases} 1 & \text{if the candidate sentence contains any} \\ & \text{pair } p \in P_j \text{ with some constraints} \\ 0 & \text{otherwise} \end{cases}$$

Going back to our initially surprising find that including stopwords in the pairs hurts performance, we attribute the difference to the fact that our features are not traditional sparse lexical features, but a relatively small number of lexical set features, and adding frequently occuring pairs to these sets renders the features virtually identical. The lexical set features rely on the fact that the member pairs are infrequent. Given the assumption that they share similar meaning in terms of the classification task, the union of the pairs becomes a meaningful and reasonably common indicator.

In addition to trying two sets of word pairs, we experimented with three variations on the constraints for allowing a sentence to be considered positive for a particular word pair.

Unigrams is the most lenient approach - consider the sentence positive if either of the words in the pair appear anywhere in it.

Unordered - consider the sentence positive only if both of the words in the pair appear anywhere in it.

Ordered - consider the sentence positive only if both of the words in the pair appear in it and furthermore, they appear in the same order as they originally did.

D. Making it Concrete - an Example

To make our description more concrete, consider the annotated claim and justification sample (4) from section I.

Our system correctly identifies the justification for this claim. Specifically, the process following the stages explained earlier is as follows.

The RST Treebank contains spans of text which are annotated with the relations PURPOSE, CAUSE and REASON. Within these, our method described in section V-A found the n-grams *because* for both CAUSE and REASON, and *in order to* for PURPOSE with high tf*-idf scores. Both of these n-grams passed our manual culling and ended in our list of indicators.

In Wikipedia, we found many cases of the words *kill* and *save* appearing in a sentence with *in order to* between them, so the pair [kill,save] made it into our pairs list for the indicator *in order to*. Similarly, the word pair [kill,killed] made it into the pairs list for the indicator *because*, since the indicator appeared in Wikipedia between the two words. These pairs are part of the *sides* list since the words in this case were on both sides of the indicator, and trivially also the *anywhere* list which is a superset.

The list features called *in order to* and *because* fire whenever the candidate sentence contains both words in the pairs [kill,save] and [kill,killed], respectively (in order or not, dependeing on the experiment), as well as other pairs found in a similar way.

The classifier learned that these two features are good enough indicators to classify the sentence as a justification. The length of the sentence in this case is not exceptional, and no pairs for other indicators were found, but these two features were enough to make this judgement.

Note that we identify two relations in the sentence: the PURPOSE relation in the first clause, and the REASON or CAUSE relation in the high-level sentence. The complexity of the sentence in terms of the number of discourse relations suggests that it contains argumentation, and our system will correspondingly have higher confidence in this sample because of the multiple positive features.

E. Experiments

Our experiment was performed on an annotated corpus of 309 LiveJournal blog threads. Out of these, we reserved 40 threads for the test set and used 269 for training. We provide results on a 10-fold cross validation of the training set (which is what we used for development) as well as on the unseen test set.

To build our data set, we take each claim and produce from it a number of data instances, each including the claim and a candidate justification sentence. Candidate justifications are all sentences which belong to an entry that is either equal to or subsequent to the entry containing the claim, and which was authored by the same poster who made the claim.[7] Positive points are those containing the actual annotated justifications, while the rest are negative. Using this method, we arrive at 6636 training instances and 756 test instances. In both data sets, approximately 10% of the points are positive.

We trained a Naive Bayes classifier (implemented with the Weka framework[8]) on combinations of the features described in the previous section, using a 10-fold stratified cross validation as the development set. Table II shows the results of the experiment. We found the results to be statistically significant using paired permutation tests on key system combinations - in particular, the best performing system (which uses *sides* with no ordering as well as the indicators) against all systems which use other pair lists or no pair features at all.

To put things in context, we also performed another experiment in which we evaluated on single sentences (as opposed to sentence-pairs). Here the task is simply to decide whether or not a sentence is a justification, for *any* claim. The sizes of the data sets in this experiment are 8508 for training and 1197 for the test set. Here we again used a Naive Bayes classifier, this time with only word pairs as features (the same features participating in our best system - *sides* with no ordering). The heuristic rules could not be applied in this experiment, as they relate to a specific claim. The baseline is simply a greedy all-positive classification.

The results are shown in Table III. They can be interpreted as the raw gain achieved by the pair features, since they only operate to identify justification sentences independent of claim in our main experiment. The heuristic rules together with the *beforeClaim* feature are the parts relating a sentence to the particular claim in the pair.

VI. Discussion

Our results show that combinations of content words can be used to predict the presence of the justification-related RST relations, and that these are helpful in identifying justifications in online discussions. This finding suggests that justification segments in the context of a dialog make use of particular rhetorical tools. Specifically, our experiments also show that it is not merely the presence of certain words that is indicative of justifcation, but specifically the presence of a discourse relation, as evidenced by the poor results for the unigram and "anywhere" pairs (as well as the bag-of-words baseline) in Table II.

The increase in performance in the pair decision task when compared to the single sentence task suggests that the presence of a claim and its location relative to the justification candidate are important and can significantly boost performance, even when only rudimentary methods are used.

Regarding indicators, it is interesting that we were able to find content words with phrases that would not in all cases be traditionally regarded as connectives; still, the location of the words with regards to the indicators is important - performance was dramatically increased when the content words came from opposing sides of the indicator.

While our best systems do not perform quite as well on the test set as they do on the cross-validation, we can attribute that to the high degree of variation of the corpus. Some variance between sets of threads is expected, and the difference in results is not so high as to warrant suspicions of overfitting. In particular, the relative contributions of the different components of our system are similar.

VII. Conclusion and Future Work

In this paper, we have addressed the issue of detecting relevance justifications in written dialogs. This is a new task.

In future work, we intend to address the following issues.

- What is the contribution of the manual culling of indicators (Section V-A)? Can we replace it with an automatic procedure (say, using tf*idf scores)? Or can we simply not do any culling and get the same results, relying on feature selection in the machine learning to filter out irrelevant features?
- By analyzing the contribution of the features, we can determine which relations and which indicators are particularly useful for identifying justifications.
- Clearly, there are cases of justification in which the justification is an atomic discourse unit, such as sample (1) in section I.

  In these cases, our approach will not contribute to accuracy since the justification contains no argumentation. We will study these justifications as a separate case, investigating how our methods can be adapted to relating an atomic justification to the claim.
- We have evaluated on our ability to relate justifications to their claims, but despite some attempts (see section IV-C), we have not actually found any features that can link the two, other than proximity in the text (and the constraint on the two being from the same writer). We will address this issue in future work, though it may be a very hard problem ("NLP-complete").

[7] Although annotators were allowed to place justifications in an earlier entry than that containing the claim, in practice no such cases exist in the corpus.

[8] http://www.cs.waikato.ac.nz/ml/weka

TABLE II
PRECISION, RECALL AND F-MEASURE OBTAINED BY THE SYSTEM IN VARIOUS EXPERIMENTS ON THE CROSS VALIDATION AND TEST SET. THE 'BASELINE' PART IN THE LATTER EXPERIMENTS REFERS TO THE HYBRID BASELINE. THE BEST SCORE AT EACH COLUMN AS WELL AS THE RESULTS FOR THE BEST SYSTEM (JUDGED BY F-MEASURE) ARE HIGHLIGHTED.

System	CV P	CV R	CV F	Test P	Test R	Test F
next sentence	**46.35**	32.44	38.17	41.67	40	40.82
heuristic baseline	28.97	**91.04**	43.95	27.16	**88.35**	41.55
hybrid baseline	41.52	54.68	47.2	31.72	45.63	40.69
bag-of-words baseline	41.37	48.57	44.68	37.5	43.69	40.36
baseline + indicators	41.52	54.68	47.2	31.72	45.63	40.69
baseline + unigrams	42.12	56.5	48.26	35.38	46	40
baseline + *anywhere* with no ordering	35.61	20.9	26.34	34.92	17.46	23.28
baseline + *anywhere* with ordering	38.17	19.81	26.08	41.67	19.84	26.88
baseline + *sides* with no ordering	42.93	61.6	50.6	**42.64**	53.4	**47.41**
baseline + *sides* with ordering	42.97	61.24	50.5	41.86	52.43	46.55
baseline + indicators + *sides* with no ordering	**43.12**	**61.81**	**50.8**	**41.86**	**52.43**	**46.55**
baseline + indicators + *sides-no-stoplist* with no ordering	42.07	58.18	48.83	37.12	47.57	41.7

TABLE III
PRECISION, RECALL AND F-MEASURE OBTAINED FOR THE SINGLE-SENTENCE CLASSIFICATION EXPERIMENT. THE BASELINE CLASSIFIES ALL POINTS AS POSITIVE.

System	CV P	CV R	CV F	Test P	Test R	Test F
baseline	11.66	100	20.89	14.75	100	25.71
sides with no ordering	30.88	48.85	37.84	30.30	40	34.48

ACKNOWLEDGMENT

This work has been supported by the Intelligence Advanced Research Projects Activity (IARPA) via Army Research Laboratory (ARL) contract number W911NF-09-C-0141. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

REFERENCES

[1] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language ann," *Language Resources and Evaluation*, vol. 39, no. 2/3, pp. 164–210, 2005.

[2] H. P. Grice, "Logic and conversation," in *Syntax and semantics, vol 3*, P. Cole and J. Morgan, Eds. New York: Academic Press, 1975.

[3] W. C. Mann and S. A. Thompson, "Rhetorical Structure Theory: A theory of text organization," ISI, Tech. Rep. ISI/RS-87-190, 1987.

[4] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, and B. Webber, "The penn discourse treebank 2.0," in *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco, 2008.

[5] S. Blair-Goldensohn, K. McKeown, and O. Rambow, "Building and refining rhetorical-semantic relation models," in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*. Rochester, New York: Association for Computational Linguistics, 2007, pp. 428–435. [Online]. Available: http://www.aclweb.org/anthology/N/N07/N07-1054

[6] E. Pitler, A. Louis, and A. Nenkova, "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*. Suntec, Singapore: Association for Computational Linguistics, August 2009, pp. 683–691. [Online]. Available: http://www.aclweb.org/anthology/P/P09/P09-1077

[7] Z. M. Zhou, M. Lan, Z. Y. Niu, Y. Xu, and J. Su, "The effects of discourse connectives prediction on implicit discourse relation recognition," in *Proceedings of the SIGDIAL 2010 Conference*. Tokyo, Japan: Association for Computational Linguistics, September 2010, pp. 139–146. [Online]. Available: http://www.aclweb.org/anthology/W/W10/W10-4326

[8] D. Marcu and A. Echihabi, "An unsupervised approach to recognizing discourse relations," in *40th Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, PA, 2002.

[9] C. Sporleder and A. Lascarides, "Using automatically labelled examples to classify rhetorical relations: An assessment," *Natural Language Engineering*, vol. 14, no. 03, pp. 369–416, July 2008.

[10] L. Carlson, D. Marcu, and M. E. Okurowski, "Building a discourse-tagged corpus in the framework of rhetorical structure theory," in *Current Directions in Discourse and Dialogue*, J. van Kuppevelt and R. Smith, Eds. Kluwer Academic Publishers, 2003.

[11] M. Moser and J. D. Moore, "Toward a synthesis of two accounts of discourse structure," *Computational Linguistics*, vol. 22, no. 3, 1996.

[12] C. Fox, "A stop list for general text," *SIGIR Forum*, vol. 24, pp. 19–21, September 1989. [Online]. Available: http://doi.acm.org/10.1145/378881.378888