Cross-Argument Inference for Implicit Discourse Relation Recognition

Yu Hong Xiaopei Zhou Tingting Che Jianmin Yao Guodong Zhou Qiaoming Zhu
Key Laboratory of Natural Language Processing of Jiangsu Province
School of Computer Since and Technology, Soochow University,
No.1 Shizi Street, Suzhou City, Jiangsu Province, China (zip code: 215006)
+86-18604601106
tianxianer@gmail.com

ABSTRACT
Motivated by the critical importance of connectives in recognizing discourse relations, we present an unsupervised cross-argument inference mechanism to implicit discourse relation recognition. The basic idea is to infer the implicit discourse relation of an argument pair from a large number of comparable argument pairs, which are automatically retrieved from the web in an unsupervised way. In this way, the inference proceeds from explicit relations to implicit ones via connective as bridge. This kind of pair-to-pair inference is based on the assumption that two argument pairs with high content similarity (i.e. comparable argument pairs) should have similar discourse relationship. Evaluation on PDTB proves the effectiveness of our inference mechanism in implicit relation recognition to the four level-1 relations. It also shows that our mechanism significantly outperforms other alternatives.

Categories and Subject Descriptors
I.2.7 [**Computing Methodologies**]: Artifical Intelligence – *Natual Language Processing (Discourse).*

General Terms
Algorithms, Measurement, Experimentation, Languages.

Keywords
Implicit discourse relation, pair-to-pair inference.

1. INTRODUCTION
The task of discourse relation recognition is to automatically predict the internal structure and logical relationship between adjacent text spans (clauses, sentences or paragraphs), such as the ones explored in our paper, Comparison, Contingency (also called Causal), Temporal and Expansion, as annotated in the Penn Discourse Tree Bank [1]. Thereafter in this paper, we limit the discourse corpus and the discourse relations to PDTB and these four relationships respectively, otherwise specified.

Corresponding Author: Guodong Zhou

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
CIKM'12, October 29–November 2, 2012, Maui, HI, USA.
Copyright 2012 ACM 978-1-4503-1156-4/12/10…$15.00.

In PDTB, a discourse relation can be either explicit or implicit. While an explicit relation is signaled directly by the use of an overt discourse connective (also called cue word) in original texts, e.g. "because", an implicit relation can only be inferred by readers according to the context of the utterance and general world knowledge, in the case that original texts don't offer the overt connective. Following two texts illustrate examples for both cases, where connective "*so*" in example (2) is missing from the text and can be only inferred by readers:

(1) He worked all night **because** the submission deadline is coming. (*Explicit relation*)

(2) He worked all night yesterday, **[Implicit=so]** slept all day today. (*Implicit relation*)

While it is easy to recognize explicit discourse relations, e.g. Pitler and Raghupathy (2008)[2] achieved the performance of more than 90% in accuracy, recognizing implicit discourse relations is very challenging, e.g. the state-of-the-art system, as described in Wang and Su (2010)[3] only received the performance of about 40% in accuracy. It is also surprising to notice that the Expansion relationship occupies more than 50% of implicit relations. That is, simply choosing the majority relationship would significantly outperform Wang and Su's system[3] by more than 10% in accuracy. The supervising low performance in current implicit relation recognition systems may be due to three major issues: the missing of connectives, the failure of finding effective linguistic features and the data sparseness problem.

For the first issue, the importance of connectives is obvious and in most cases, the discourse relationship can be determined correctly by the occurrence of a connective, as confirmed in Pitler and Raghupathy (2008) [2].

For the second issue, so far there are no previous studies on systematically exploring why implicit relations don't need the cue words (i.e. connectives). Our observation suggests that implicit relations often occur in parallelism, antithesis, polite formula, rhetorical question, idiom and dialogism, indicating the habitually default cue of well-known and acknowledged discourse relations, or that of relations in casual conversations.

This makes it quite difficult to recognize implicit discourse relations by using the existing linguistic features. Example (3) shows the difficulty to use syntactic features to recognize implicit relations in parallelism due to the failure for syntactic parse trees to reflect the rhetorical structure of parallelism, while example (4) shows the difficulty of predicate word-pair features in dialogism

& rhetorical question: What can the keyword pair "*go-Go*" do for the discourse analysis?

(3) Current college students study 24 hours a day, **[Implicit=and]** 7 days a week, **[Implicit=but]** two weeks a year. (*Parallelism*)

(4) You wanna go? **[Implicit=then]** Go now. (*Dialogism & Rhetorical question*)

Motivated by the critical importance of connectives in recognizing explicit discourse relations, this paper proposes a cross-argument inference mechanism to overcome the three major issues in recognizing implicit relations. Here, an argument (abbr., Arg) denotes a text span by the definition of PDTB. The basic idea is to infer the implicit discourse relation of an argument pair from a large number of comparable argument pairs, which are automatically retrieved from the web in an unsupervised way. This kind of pair-to-pair inference is based on the assumption that two argument pairs with high content similarity (i.e. comparable argument pairs) should have similar discourse relationship. In this way, the inference proceeds from explicit relations to implicit ones via connective as bridges. Example (5) shows two argument pairs meeting the requirements of the cross-argument inference:

(5) Arg1: Bush beats McCain,
Arg2: **and** wins the election. (*Explicit relation*)

Arg1: Obama beats McCain,
Arg2: **[Implicit=and]** winning the election. (*Implicit relation*)

Obviously, above three major issues can be well addressed by our cross-argument inference mechanism, i.e. through employing connective as bridge to achieve the inference from explicit relations to implicit relations, avoiding the intricate linguistic or rhetorical analysis via direct discourse relation mapping between comparable argument pairs and data sparseness via retrieving comparable argument pairs from the large-scale web data.

The rest of the paper is organized as follows. Section 2 overviews related work. Section 3 introduces PDTB. Section 4 describes our cross-argument inference mechanism for implicit discourse relation recognition. Section 5 shows the experiments and discussion. Finally, we conclude our work in Section 6.

2. RELATED WORK

Similar to explicit relation recognition, all the existing studies on implicit recognition adopt a supervised learning framework with focus on exploring various kinds of linguistic features.

Pitler and Louis (2009) [4] explored several linguistically informed features, including polarity tag, Levin verb class, length of verb phrase, modality, context, and lexical features. They worked with Penn Discourse Treebank [1], the largest available annotated corpora of discourse relations, showing performance increases over a random classification baseline.

Lin and Kan (2009)[5] inherited this theme and expended the contextual features by using the dependency patterns and the possible correlations between pairs of discourse relations. Their classifier considered the context of the two arguments, as well as the arguments' internal constituent and dependency parses, showing an accuracy of 40.2% on PDTB.

Wang and Su (2010) [3] focused on syntactic knowledge via a tree kernel based approach. They achieved the performance of 40.0% in accuracy on PDTB, slightly outperforming a flat-syntactic-path based approach.

Using the RST corpus (Carlson and Marcu, 2001)[6], which contains 385 Wall Street Journal articles annotated according to the Rhetorical Structure Theory (Mann and Thompson, 1988) [7], Soricut and Marcu (2003)[8] exploited the fact that many of the useful features in explicit discourse relation recognition, syntax in particular, would not be applicable to the analysis of implicit relations that occur intersententially. While their earlier work (Marcu and Echihabi, 2002) [9] showed that the co-occurrence probability of word pairs that extracted from two text spans respectively is useful for detecting implicit discourse relation. Saito and Yamamoto (2006) [10] extended this theme, to show improvements by phrasal patterns extracted from text span pairs. These imply the feasibility of statistical methods in implicit discourse relation recognition.

Meanwhile, Wellner and Pustejovsky (2006) [11] showed that the features of discourse connectives and the distance between the two text spans have the most impact to both explicit and implicit discourse relation analysis. Their experiments on GraphBank (Wolf and Gibson, 2005) [12] achieved 81% accuracy in relation sense disambiguation. However, their system may not work well for implicit relations alone, as the two features only apply to explicit relations: implicit relations don't have discourse connectives and the two text spans of an implicit relation are usually adjacent to each other. Besides, the GraphBank annotations don't differentiate between implicit relations and explicit ones, so it is difficult to verify success for the implicit relations.

Nevertheless, Zhou and Xu (2010)[13] adopted a bigram language model to predict implicit discourse connectives between arguments, which improved their supervised binary classification (either positive or negative relation) on almost four 1-level implicit discourse relations. Besides, they built an unsupervised classification, which mapped each predicted connective to its most frequent relation sense in training data and used the sense to directly appoint the implicit discourse relation. However, this approach only achieved improvements for Contingency and Temporal relations with accuracies of 70.79% and 70.51% respectively, although it can reach more than 97% accuracy for every relation independently when using gold-truth implicit connectives. This shows the shortcoming of connective prediction.

3. PENN DISCOURSE TREE BANK

For our experiments, we use the Penn Discourse Treebank [1], the largest resource of annotated discourse relations. The annotation covers the same one million word Wall Street Journal corpus used for the Penn Treebank [14].

The PDTB adopts the connective-argument view of discourse relations, where a discourse connective (e.g., *because*) is treated as a trigger that takes two text spans as its arguments. The argument that the discourse connective structurally attaches to is called Arg2, and the other argument is called Arg1. Example (6) shows two arguments triggered by the connective "*because*".

(6) **Arg1**: He is very tired,
Arg2: **Because** he played football all morning.

(Contingency-wsj_*ID*)

The PDTB is the first corpus to systematically identify and distinguish explicit and implicit discourse relations, allowing us to concentrate solely on the implicit relations. By definition, an explicit relation is triggered by the presence of a discourse connective which occurs overtly in the text. The discourse connective can essentially be viewed as a discourse-level predicate which takes two clausal arguments. Example 6 shows an explicit Contingency relation triggered by the discourse connective "*because*", where the last line shows the relation type and the file (*ID* indicates file number) in the PDTB from which the example is drawn. The corpus recognizes 100 such explicit connectives and contains annotations for 19,458 explicit relations.

Implicit relation is inferred by the reader but not marked by an overt discourse connective in the text. In this case, for a pair of adjacent spans, the annotator was asked to provide a connective that best captured the inferred relation. Example (7) shows an implicit relation, where the annotator inferred a Contingency relation and inserted an implicit connective "*so*" (i.e., the original text does not include "*so*"). There are a total of 16,584 implicit relations annotated in the corpus.

(7) **Arg1**: You look tired.
Arg2: **[Implicit=So]** Better to have a rest.

(Contingency-wsj_*ID*)

In addition to discourse relations and their arguments, the PDTB also provides the senses of each relation (Miltsakaki and Livio, 2008). The tagset of senses is organized hierarchically into three levels. The top level consists of four major relation classes: Temporal, Contingency, Comparison, and Expansion. For each class, a second level of types is defined to provide finer semantic distinctions. A third level of subtypes is defined for only some types to specify the semantic contribution of each argument. In this paper, we focus on implicit relation recognition to the Level 1 classes, as until recently, the classes could be predicted with only 40.2% in accuracy at best (Lin and Kan, 2009) [5].

4. CROSS-ARGUMENT INFERENCE

In this section, we firstly give the framework of our cross-argument inference method, and then approach the issues of comparable argument pair mining, relation sense disambiguation and pseudo-cue filtering. The first is to detect the comparable argument pairs to form the basis of the inference; the second is to disambiguate the relation senses of explicit connectives in the mapping process (e.g. discrimination between Contingency and Temporal senses of the connective "since"); the third is to shield the inference from the misleading cues caused by the unbalanced distribution of discourse relations (averagely 42.45% Expansion relations but only 12.89% Temporal in PDTB).

The cross-argument inference framework predicts the implicit relation for a given relational argument pair with four basic steps (see Table 1). To perform the pair-to-pair inference from explicit relations to implicit ones, the first step only focuses on mining the comparable argument pairs, which have explicit connectives. On the basis, the second and third steps attempt to select the optimal comparable argument pairs to explore the most possible connective (cue word) for the inferred argument pair. The last step implements the relation mapping with the help of the normally well-defined correspondence between relation senses and connectives (e.g. the connective "because" signals the "Expansion" sense with 100% probability in PDTB).

Table 1: The cross-argument inference framework

Target: **Arg1** [implicit=?] **Arg2**
Task: Predict the implicit **relation**
Inference Procedure
Step 1: Mine the candidate comparable argument pairs for the target from Web; Step 2: Calculate the similarity between the target and each candidate in content, and rank the candidates based on the similarity; Step 3: Select top N' candidates in the ranked list as the comparable argument pairs, and insert the most frequently occurring connective in the comparable pairs into the slot [implicit=?]; Step 4: Map the connective to a relation sense by which to signal the implicit relation of the target.

The proposal that comparable argument pairs can be successfully mined is ensured by the existence of a massive amount of reproduced and adapted texts in different information medium. Actually, once texts represent the same knowledge, even if the texts are created independently, they can still form the comparable resources because inevitably involving the same components of the knowledge as well as their inherent relations, e.g. every *attack* instance involves the element of *hurt* or even *death*. Example (8) shows two instances of the knowledge "*Attack*", which form the comparable resources by the same components "*attack*", "*hurt*", "*death*" and their Contingency and Expansion relations.

(8) **Instance 1**: Our soldier *attacked* the terrorists, **caused** 3 *dead*, **and** 1 *injured*.
Instance 2: The terrorists *attacked* our base, 1 *dead*, 3 *injured*.

(*cause*: Contingency; *and*: Expansion)

Perfect relation mapping can only be ensured by the one-to-one (viz., unambiguous) correspondence between connectives and relation senses. However, despite few, there exist exceptive connectives often signaling multiple relation senses, e.g. the connective *since* has Contingency and Temporal as its senses with almost equal probability in PDTB (Pitler and Raghupathy, 2008) [2]. In this case, the relation mapping should be supplemented by relation sense disambiguation (see section 4.3). And according to ambiguity identification for connectives, the cross-argument inference framework decides whether to perform one-to-one relation mapping or mapping with relation sense disambiguation.

Our inference method regards the most frequently occurring connectives in comparable argument pairs as cues to signal the inferred implicit relations. However, the frequency statistics will be unavoidably confused by the unbalanced distribution of discourse relations. Table 2 shows the distribution between the four main relation classes and their type of realization (implicit or explicit) in PDTB. It can be found that the Expansion relations are distributed predominantly in both explicit and implicit, having the largest number of discourse instances connected by such relation. On the contrary, the Temporal relations are always distributed sparsely, particularly in implicit with only 5.73% occurrence. Thus, if this is the natural distribution in overall linguistic

resources, our frequency based connective selection will bring in the connectives of more widely distributed relations, to make noises (pseudo-cues) in recognizing sparse relations. Therefore, the connective selection should be supplemented by an additional treatment of pseudo-cue filtering.

Table 2: Discourse relation distribution in explicit/implicit classes in the PDTB

Class	Explicit (%)	Implicit (%)
Comparison	5590 (28.73%)	2505 (15.10%)
Contingency	3741 (19.23%)	4261 (25.69%)
Temporal	3696 (18.99%)	950 (5.73%)
Expansion	**6431 (33.05%)**	**8868 (53.47%)**
	Total = 19,458	Total = 16,584

4.1 Mining Comparable Argument Pairs

In this paper, we propose a web-driven comparable argument pair mining method for the unsupervised cross-argument inference. It includes two basic steps: the first is to use information retrieval with a regular expression constrained query to roughly search possible comparable argument pairs (candidates); and the second is to select the most similar candidates with the tested argument pairs in content to generate the final comparable corpus.

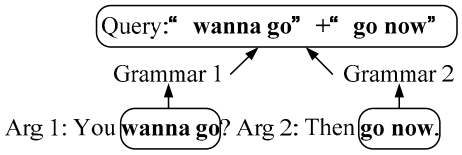

Figure 1. Example of query structure

We perform the information retrieval on the search engine Google. The query is generated by the use of two grammars extracted from the tested argument pair: one grammar (*PreGram*) from the prepositive argument, and another (*PosGram*) from the postpositive. And the query follows the regular expression “*PreGram*”+“*PosGram*”, within which the double quotes and the plus sign are both necessary parts, as shown in Figure 1. A query expressed as “X”+“Y” in Google can ensure that all search results (especially snapshots) contain X and Y simultaneously but in different spans.

For each tested argument pair, we use all possible combinations of its “*PreGram*” and “*PosGram*” to generate queries, and use snapshots of top *N* (*N*=500) search results as resources for extracting candidate comparable argument pairs. We select the snapshots but not full texts as the resources because the former provide closers comparable pairs in short texts (averagely 3 to 4 spans), helping to obtain adjacent comparable pairs efficiently. And obviously the adjacent pairs have more useful relations for our inference than estranged pairs.

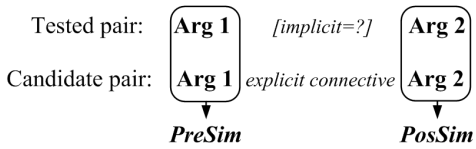

Figure 2. Calculation of *PreSim* and *PosSim*

To support Explicit-to-Implicit inference, we only select the adjacent pairs which have explicit connectives to generate the candidate comparable corpora. Besides, if a query can’t contribute more than *n* (*n*=5) eligible comparable pairs (viz. adjacent pairs with an explicit connective) in top *N* search results, we ignore the query along with its comparable pairs. This is because the sparse eligible pairs imply invalidity of the query in exploring discourse relations, and thus the pairs may occur randomly and very possibly become noises in the candidate comparable corpora.

On the basis, we measure the comparable similarity in content between the candidates and the tested argument pair, and select the *N′* (*N′*=20) most similar candidates as the final comparable corpora. In the measurement, the similarity of prepositive arguments (*PreSim*) and that of postpositive (*PosSim*) are calculated respectively (see Figure 2), and the whole similarity is the sum of *PreSim* and *PosSim*, divided by the exponentially smoothed absolute value of their difference:

$$Sim = \frac{PreSim + PosSim}{2 \cdot e^{|PreSim - PosSim|}} \quad (1)$$

where, the exponential function is to ensure nonzero denominator, and the number 2 normalizes the whole similarity. We calculate the similarity like this is to ensure unbiased contributions of *PreSim* and *PosSim* to comparable detection. The bias is measured by the difference between *PreSim* and *PosSim* (viz. |PreSim-PosSim| in the denominator). Thus a good comparable pair should have high similarities *PreSim* and *PosSim* simultaneously with the tested pair. We calculate the *PreSim* and *PosSim* using VSM-based Cosine metric, and use bi-gram as the grammars in query generation and argument representation. Besides, the values of *n* (*n*=5), *N* (*N*=500) and *N′* (*N′*=20) are fine-tuned on all explicit arguments of PDTB v2.0 with a sub-system of predicating connectives.

4.2 Relation Sense Disambiguation

In our inference method, the relation sense mapping can be confused by ambiguous senses of some connectives. In this paper, we propose a local rigorous connective based relation sense disambiguation method.

The sense disambiguation uses unambiguous connectives in comparable corpora (local rigorous connectives) to infer the senses of ambiguous connectives in three basic steps: given an ambiguous connective, the first is to cluster comparable argument pairs; the second is to measure the divergence between local rigorous connectives and the ambiguous connective in the clusters; and the third detects the close rigorous connectives and use their most frequently occurring sense as the sense of the ambiguous connective. Therefore, the disambiguation actually performs a sense inference among similar comparable pairs in content.

We cluster the comparable corpora using the agglomerative hierarchical clustering algorithm (Davidson and Ravi, 2005) [15], by which to generate the cluster tree with the comparable pairs as leaf nodes, where each node is represented by its connective. On the basis, we calculate the divergence by the path length between leaf nodes. We statistically model the close rigorous connective based sense inference as:

$$P(s) = \frac{P_s}{log(\overline{div_s})} \quad (2)$$

where, $P(s)$ is the probability of sense s acting as the sense of the ambiguous connective in the comparable corpora, P_s is the prior probability of s, $\overline{div_s}$ denotes the average divergence between the ambiguous connective and rigorous connectives of sense s in the cluster tree, and the logarithmic function smoothes the sharp difference of divergences.

The sense inference model tends to recommend senses of closer rigorous connectives when the prior probability P_s closes to 0.5. Such probability (P_s=0.5) means corresponding relation senses are the most ambiguous, e.g. the 52.17% probability of sense Contingency of connective "*since*" in Table 3. On the contrary, when P_s is much high (indicating unambiguous), e.g. the 95.99% probability of Contingency of "*moreover*" in Table 3, the model normally can't influence the predominant senses of connectives with the help of logarithmic function. Therefore, it can facilitate sense ascertainment of very ambiguous connectives in the meanwhile avoid confusing senses of approximate rigorous connectives.

Table 3: Percentage of occurrences of connective in its predominant sense (prior probability)

Comparison: *while* (66.07%), *But* (97.19%), *yet* (97.03%), *still* (98.42%), *however* (99.59%), *although* (99.70%), *though* (100.00%)
Expansion: *in fact* (92.68), *indeed* (95.19%), *and* (96.83%), *or* (96.94%), *instead* (97.32%), *unless* (98.95%), *also* (99.94), *for example* (100.00%), *in addition* (100.00%), *moreover* (100.00%), *for instance* (100.00%), *separately* (100.00%)
Contingency: *since* (52.17%), *if* (95.99%), *because* (100.00%), *so* (100.00%), *thus* (100.00%), *as a result* (100.00%)
Temporal: *meanwhile* (48.70), *as* (70.26%), *when* (80.18%), *until* (87.04%), *then* (93.24%), *once* (95.24%), *later* (98.90%), *after* (99.65), *before* (100.00%)

More directly, we define connectives with more than 90% prior probability of occurrence in their predominant senses as the rigorous connectives, and those less than the probability as ambiguous ones. The sense disambiguation only proceeds when meeting the defined ambiguous connectives. We show all explicit connectives that appear more than 50 times in the PDTB, as well as the prior probability in their predominant senses (see Table 3). It can be found only a few connectives (e.g. while, since, meanwhile, etc) need to be disambiguated, and thus most connectives can be used as rigorous connectives to support the sense inference. In our Explicit-to-Implicit relation inference, only explicit connectives are used in sense mapping. So we divide explicit connectives of PDTB into rigorous (82 in total 102 explicit connectives) and ambiguous (the rest 20) to perform disambiguation.

4.3 Filtering Pseudo-Cues

As discussed in section 4.1, the unbalanced distribution of discourse relations will make our frequency based connective selection easily bring in the connectives of more widely distributed relations, resulting in noisy connectives (pseudo-cues) in sense mapping for sparsely distributed relations. For example, the word "*and*" is an Expansion connective the most frequently occurring in most linguistic resources. However, the spans, which are connected by the connective "*and*", don't always have Expansion relation (See Example 9). In this case, the connective "*and*", as a noise, brings in a pseudo-cue for the discourse relation prediction.

(9) They read lots of books, **and later** made a plan.
(Original connective: *and* - Expansion)
(Latent connective: *later* - Temporal)

By analyzing the connectives in comparable corpora for section 23 of PDTB v2, we found that most pseudo-cues are the connectives which indeed don't trigger discourse relations. Example (9) shows a Temporal relation but has an Expansion connective, where the real trigger should be the word "*later*" but not the connective "*and*".

Actually, this is very common because connectives also have the function of expressing mood (i.e. modal particle), e.g. coherence, pause, emphasis, etc. And, in these cases, discourse relations are normally triggered by other latent connectives, e.g. "*later*" in Example (9). However, although there are many latent connectives, most of them are never publicly released, especially in PDTB.

The latent connectives can be divided into two classes as follows:

- **Connective Triggers (CTs)**: They are the connectives (e.g. "*later*" in Example 9) which trigger discourse relations but don't connect argument pairs. They normally co-occur with pseudo-cues (e.g. "*and*" in Example 9) (Note: Pseudo-cues are the connectives which connect argument pairs but don't trigger discourse relations)

- **Functional Triggers (FTs)**: They aren't connectives but have the function of triggering discourse relations (e.g. "*lead to*" triggers Contingency relations). (Note: FTs haven't publicly released so far). Table 4 shows several FTs that frequently occur in PDTB corpus (The whole FTs are listed in the section of our experiments).

Table 4: Examples of Functional Trigger

Contingency: *cause*, *make*, *lead to*, *result in*, *bring about*
Expansion: *restatement, to illustrate, including, coupled with*
Comparison: *compared with, similar to*
Temporal: *last year, recently, sometimes, immediately, suddenly*

In this paper, the comparable argument pairs, which involve both an original connective and a latent connective (either connective trigger or functional trigger), are regarded as the samples whose discourse relations are at risk of being misled by pseudo-cues. To reduce the negative influence of the samples in Slot Prediction [implicit=?] (see Step 3 in Table 1) and Relation Sense Mapping (see Step 4 in Table 1), we adopt four methods of pseudo-cue filtering:

- **Rough Filtering**: directly filter out the samples from the set of comparable argument pairs. Hereafter, the samples will not be used to predict the connective of the slot [implicit=?], perform sense mapping, disambiguate relation sense and any part of cross-argument inference.

- **Agent Filtering**: ignore the original connectives of the samples, and use the latent connectives as agents to perform the inference. E.g. in example (9), the latent connective "*later*" is actually used to predict the slot [implicit=?] but not the original connective "*and*".

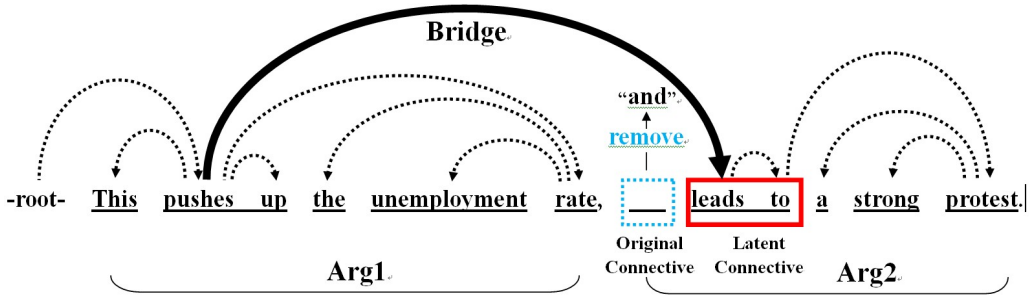

Figure 3. An example of Sneak Filtering for Functional Trigger ("*and*"-Expansion; "*lead to*"-Contingency)

- **Sneak Filtering (for FTs)**: remove original connectives from the samples, and then parse the samples (Stanford dependency parser[1] is used). In a dependency tree, if a FT has the bridge connecting Arg1 and Arg2, it will be regarded as the true connective and used to predict the slot [implicit=?], else the original connective will be used insusceptibly. See the example of sneak filtering in Figure 3, where after removing the original connective "*and*", the only bridge (dependency arc connect two Args) points to the FT "*lead to*", which will be used as the true inference cue (Note: when two Args have no FT or have but triggering a clause but not the bridge between the Args, the sneak filtering is unavailable).

- **Sneak Filtering (for CTs)**: remove original connectives from the samples, and then parse the samples. In a dependency tree, if a CT has a direct dependency arc with the word which bridges two text spans, it will be regarded as the true connective, else the original connective will be used insusceptibly. See the example of sneak filtering in Figure 4, where after removing the original connective "*and*", the only bridge points to the word "*go*", and the CT "*later*" has a direct dependency arc with it, according to which, sneak filtering regards the CT as the true inference cue.

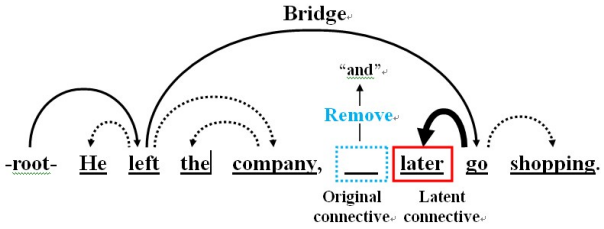

Figure 4. An example of Sneak Filtering for Connective Trigger ("*and*"-Expansion; "*later*"-Contingency)

Sneak filtering aims to determine whether a latent connective can bridge two arguments instead of an original connective when the two connectives occur in the same sentence. If yes, the latent connective will be used as the true inference cue in our discourse relation recognition scheme. This is under the hypothesis that if a content word (Functional Trigger) or a connective which depends on the content word (Connective Trigger) has reflected the semantic or logistical relation between text spans, the original connective will not trigger the main discourse relation or even just a modal particle.

1 http://nlp.stanford.edu:8080/parser/index.jsp

In comparison, the Rough Filtering avoids using the comparable argument pairs, which have ambiguous relations, to infer implicit discourse relations. The Agent Filtering utterly biases toward applying discourse relations of latent connectives to the inference. But the Sneak Filtering exposes the possible pseudo-cues and selectively replaces those using latent connectives (including FTs and CTs) before the inference.

5. EXPERIMENTATION

5.1 Experimental Settings

We experiment on the four top-level discourse relations Temporal (abbr., *Tem.*), Contingency (*Con.*), Expansion (*Exp.*) and Comparison (*Com.*) of PDTB v2 corpus. We use implicit instances in sections 23-24 for testing. However, unlike previous work, which uses instances in the rest of sections for training, we don't need training corpus for our unsupervised inference. The parameters n (5), N (500) and N' (20) are fine-tuned from the optimization process of comparable corpora mining on all explicit instances. Table 5 shows the distribution of test instances in the top-level relations.

Table 5: Distributions of instances (argument pairs)

Relation	*Exp.*	*Con.*	*Com.*	*Tem.*
Instance	606	338	208	40
	(50.80%)	(28.40%)	(17.40%)	(3.40%)

We implement four cross-argument inference systems to respectively test comparable pair mining, sense disambiguation and pseudo-cue filtering:

- **System 1**: Perform the inference using candidate comparable corpora retrieved from Google;

- **System 2**: Optimize the comparable corpora of System1 using comparable similarity measurement (see equation (1));

- **System 3**: Optimize the sense mapping of System 2 by using local rigorous connective based relation sense disambiguation;

- **System 4**: Improve System 3 by using the best pseudo-cue filtering method (Sneak Filtering). We will exclusively compare the performances of the filtering methods in Section 5.3.

Besides, we reproduced and tested Zhou et al (2010)'s method[13] on the same corpus (viz., sections 23-24 of PDTB v2) which use a language model to mine consistent expression patterns, by which to predict connectives and further implicit relations. It is a

good statistical method to compare with our inference method. And we also reported the performance of Wang et al (2010)'s method [3], which improves the syntactic-feature based discourse relation classification with the tree kernel. We compared it with our method (both being tested on sections 23-24) to verify the feasibility of statistical modeling in discourse relation analysis. The performance is evaluated as:

$$Accuracy = \frac{TruePositive + TrueNegative}{All} \quad (3)$$

This metric is specially used to evaluate individual relation recognition, where an instance is inferred to be a target relation (*positive*) or not (*negative*). When evaluating the accuracy for all relations (*four-way*), the *TrueNegative* always equals to 0, to generate a whole accuracy (i.e. percentage of correctly inferred instances).

Table 6: Accuracy of our systems

	System1	System2	System3	System4
Exp. vs. Other.	43.04%	57.95%	58.09%	55.98%
Con. vs. Other.	45.53%	68.15%	69.72%	**72.51%**
Com. vs. Other.	69.25%	81.12%	82.39%	**85.16%.**
Tem. vs. Other.	88.29%	95.30%	96.13%	**96.86%**
Four-way	38.03%	52.85%	54.61%	**57.55%**

5.2 Main Results

We firstly inspect the performance of our systems on recognizing an individual relation. Table 6 lists the accuracy for each of the target relations. The worst performance of system 1 illustrates our web-driven approach with comparable corpora mining initially offers rough resources for comparable relation detection. System 2 optimizes the initial corpora through effective comparable similarity measurement, achieving substantial improvements on overall target relations (22.62% at best). System 3 further slightly improves System 2 (1.57% at best). System 4 achieves the optimal four-way performance and considerable improvements on Contingency and Comparison (2.79% at best), but shows a little worse on Expansion than System 3.

Table 7: Number of correctly inferred instances

	System2	System3	System4	Total
Exp.	346	346(↑0)	319(↓27)	606
Con.	151	162(↑11)	195(↑33)	338
Com.	113	122(↑9)	151(↑29)	208
Tem.	20	21(↑1)	21(↑0)	40

We analyze the results of our systems for each of the target relations, to explore the nature of the performance of System 3 and System 4. Table 7 lists the number of correctly inferred instances by System 3, which only offers 21 new correct inferences in total 1192 test samples. This might suggest that our sense disambiguation doesn't work very well. But actually only 138 test samples have ambiguous connectives and 104 have been correctly inferred before System 3 (by System 2), so our disambiguation method has a recall of 61.76% (21/34) in the rest 34 ambiguous relations.

Besides, Table 7 shows System 4 loses 27 Expansion instances, resulting in the reduction in accuracy on the relation. On the contrary, System 4 brings in more correct inferences for Contingency and Comparison. This sharp contrast illustrates our pseudo-cue filtering tends to detect the pseudo-cues derived from Expansion connectives, and thus shield the inference on Contingency and Comparison from noisy connectives of Expansion, but simultaneously filter some other originally correct Expansion connectives. This, to some extent, is unavoidable because Expansion connectives distribute so widely that their risk of being filtered wrongly is increased. Table 8 shows the major connectives filtered as pseudo-cues, which are mostly Expansion connectives.

Table 8: Connectives filtered as pseudo-cue (*num*)

Expansion: *and* (1,891), *or* (698), *instead* (155), *nor* (96), *indeed* (59), *in fact* (34), *also* (29)
Comparison: *but* (304), *yet* (56), *while* (19)
Temporal: *once* (86), *later* (62), *then* (51)

Table 9: The whole latent connectives occurred in PDTB (including Connective Triggers and Functional Triggers)

Contingency	**The 3 most frequent in PDTB: "cause", "make", "lead to".** *due to, plainly, in that case, arouse, evoke, induce, provoke, generate, bring about, contribute to, result from, result in, set off, stem from, be attributed to, arise from, thanks to, in order to, such that, in this way, cause, make, lead to.* **(Sequence number: Con1-Con23)**.
Expansion	**The 4 most frequent in PDTB: "restatement", "such as", "even", "including".** *a case in point, as an illustration, as an example, just as, just like, namely, to illustrate, to demonstrate, chiefly, markedly, particularly, for one thing, in this case, along with, equally, apart from, similar to, coupled with, aside from, barring, excluding, outside of, by the way, incidentally, above all, as a matter of fact, most important, obviously, to be sure, undoubtedly, actually, emphasis, no doubt, in summary, to conclude, to sum up, conclude, as a rule, as usual, for the most part, ordinarily, usually, by and large, in any case, in any event, on the whole, in the long run, on balance, in my opinion, restatement, such as, even, including.* **(Sequence number: Exp1-Exp53).**
Comparison	**The 5 most frequent in PDTB: "than", "like", "compared with", "despite", "similar to".** *similar to, compared with, unlike, after all, same, in spite of, except for, in comparison, different from, worst of all, alike, together with, admittedly, than, like, compared with, despite.* **(Sequence number: Com1-Com21).**
Temporal	**The 4 most frequent in PDTB: "last year", "recently", "sometimes", "immediately".** *suddenly, the moment, for now, at first, in time, after that, all of a sudden, in the first place, till, after a while, for the time being, at that moment, the next week, in an hour, the next step, in a few days, at present, from now on, last year, recently, sometimes, immediately.* **(Sequence number: Tem1-Tem22).**

5.3 Filtering Performance Comparison

To support the pseudo-cue filtering, we manually collected 165 Functional Triggers (FTs) which are all content words (verbs, adjectives, adverbs) or their corresponding phrases. After expanding the triggers according to the hyponymy of WordNet [16], we finally obtained 396 FTs, including 98 Contingency, 179 Expansion, 51 Comparison and 68 Temporal. We regard the FTs as the key latent connectives. Besides, we expanded the 101 connectives of PDTB by adding 18 new connectives (e.g. "*than*"), which all are conjunctions or prepositions and their combinations. When the connectives co-occur in the same sentences, the connectives, which don't connect text spans, will be automatically regarded as latent Connective Triggers (viz., CTs, see Section 4.3). There are totally 186 FTs and 15 new connectives occurring in PDTB. Table 9 lists 119 of them which occur more than 5 times.

We inspected the distributions of Functional Triggers (FTs) respectively in implicit and explicit relations (see Figure 5), by which to validate the feasibility of using the FTs as the cues of our relation inference. Figure 5 shows the FTs more frequently occur in implicit relations than in explicit relations, especially the most frequent FTs give extremely obvious differences. It to some extent illustrates that when there aren't explicit connectives, the FTs can instead express the discourse relations.

However, there are few implicit argument pairs having FTs (51% cases in PDTB corpus), especially the FTs which indeed connect the argument pairs are sparse (only 31% cases in the sections 23-24 of PDTB v2). Thus it is hard to directly use FTs to predict implicit relations (by sense mapping in Table 9). An alternative way is to use the distributions of FTs in a large scale of comparable argument pairs (mined from Web data) to statistically predict the most possible FT and perform sense mapping. However, the cases that FTs solely occur in the argument pairs are fewer than the cases that FTs co-occurred with explicit connectives. This may derived from that people normally avoid the informal usage of FTs in directly connecting text spans. All in all, the following reasons motivate us to use FTs to filter pseudo explicit connectives (viz., pseudo-cues) but not solely use it to perform relation prediction:

- FTs has the function to express discourse relations;
- FTs normally co-occurred with explicit connectives but sparsely solely occur;

We performed the pseudo-cue filtering methods (see Section 4.3) and use them respectively to improve the system 3 (viz., our cross-argument inference system after optimizing comparable corpora and sense mapping). The performances are shown in Table 10. Within the table, the baseline 1 detects the target

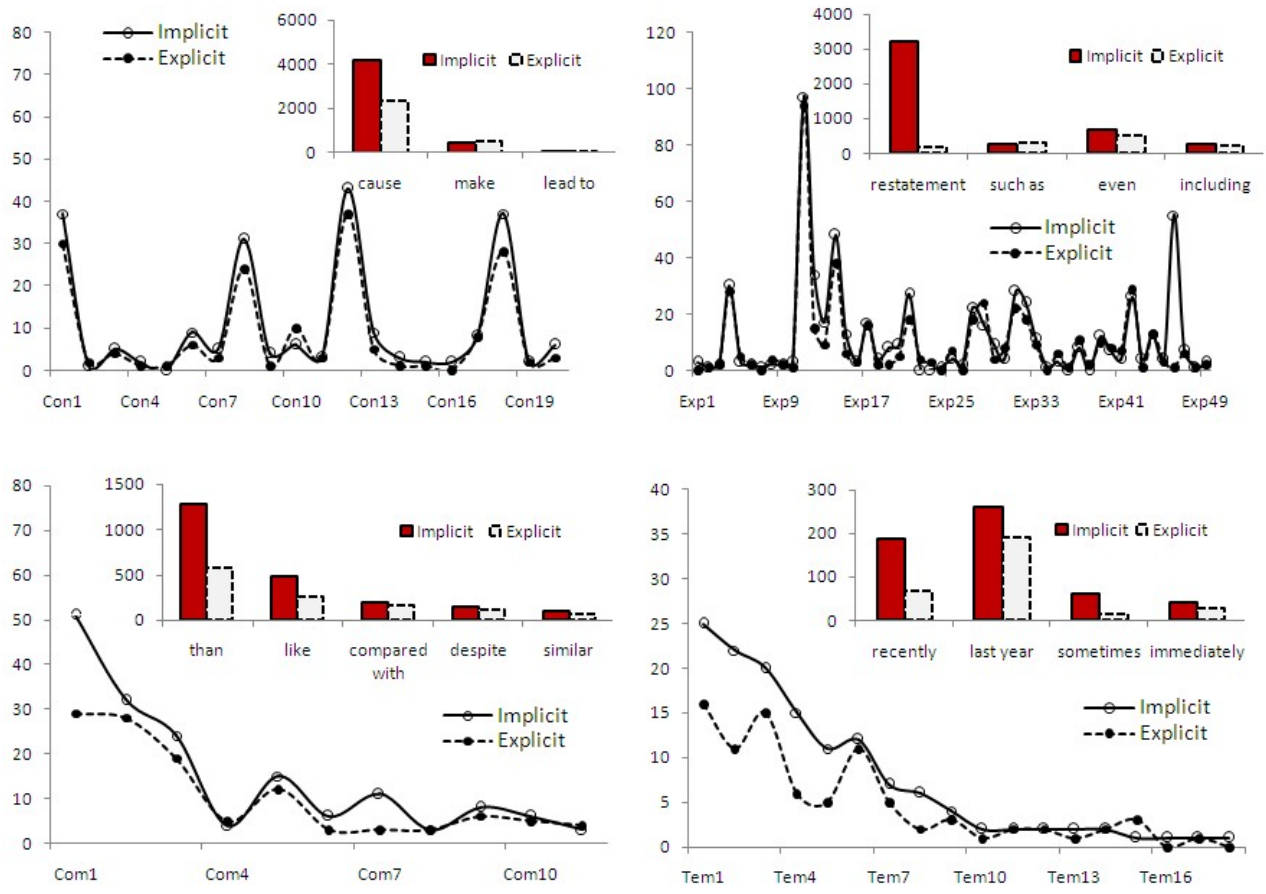

Figure 5. The distributions of Functional Triggers (FTs) in implicit and explicit relations (The vertical axis indicates the occurrence frequency of FTs, the horizontal axis indicates the sequence number of FTs which have been mentioned in Table 9)

implicit argument pairs (in the sections 23-24 of PDTB v2) which involve FTs and uses the FTs to directly predict relations (by using the sense mapping in Table 9), and other targets maintain the original output of System3. And the baseline 2 detects the comparable argument pairs (in the mined comparable corpora) which involve FTs and uses the FTs to directly predict relations. For System 3, the Rough filtering removes all comparable pairs which involves both original connectives and latent connectives (including CTs and FTs). The Agent filtering uses the latent connectives to replace the original connectives. Sneak filtering selectively performs the replacement according to the dependencies of the latent connectives (see Section 4.3).

Table 10: Filtering Comparison
(The performance of System 3 is 54.61% on Four-way)

Method	Accuracy
System 3+ **Filtering Baseline 1**	28.50% ↓
System 3+ **Filtering Baseline 2**	29.25% ↓
System 3+**Rough**	52.62% ↓
System 3+**Agent**	49.34% ↓
System 3+ **Sneak (CTs)**	**55.21%** ↑
System 3+ **Sneak (FTs)**	**56.92%** ↑
System 3+ **Sneak (CTs + FTs)**	**57.55%** ↑

It can be found that the three methods of Sneak Filtering all improve the performance of System 3, especially the Sneak which considers both FTs and CTs. And after analyzing the experimental data, we found the sneaks should bring in more improvements if the dependency parser is more precise.

In contrast, the Agent filtering and the Rough filtering reduce the performance of System 3, especially to directly replace original connectives by latent connectives in Agent filtering brings in much more performance drawback. So compared with discarding some uncertainty, to unlimitedly replace original connectives is at the risk of bringing in more noises. The baseline 1 and 2 show the worst performances. But in fact the baseline 1 should achieve an approximately 31% accuracy because the percentage of the FTs, which connect the argument pairs, is about 31%. This implies some incorrect relation sense mapping or ambiguous senses of our FTs. Besides, we find the baseline 1 can be improved if we further expand the set of latent connectives, e.g. adding the combination of synonyms (e.g. "…*beauty*…*pretty*…" - Expansion) and that of antonyms (e.g. "…*love*…*hate*…" - Comparison).

Therefore, to improve pseudo-cue identification and filtering, it is necessary to deal with the following issues:

- To further expand the set of latent connectives;
- To disambiguate multiple senses of latent connectives;
- To improve the precision of dependency parser.

5.4 Inference Performance Comparison

We compare our best system with that of Wang et al's Tree kernel method[3] and Zhou et al's language model based unsupervised method[13]. Besides, we use the majority relation class (see Table 4) as the baseline, where all instances are classified as Expansion, yielding an accuracy of 50.80%. Table 11 lists the accuracy for all relations (four-way). Our method (System 4) shows the best performance, better than Wang et al's and Zhou et al's respectively by 17.55% and 16.20% respectively, although only better than the baseline by 6.75% in accuracy.

Table 11: Overall results for tested methods

Method	Accuracy
Tree Kernel (Wang et al's)	40.00%
Language Model (Zhou et al's)	41.35%
Our best (System3 + Sneak (FTs+CTs))	57.55%
Baseline	50.80%
Human 1	**61.29%**
Human 2	**59.78%**

Although we use our best system (System 4) to compare with Wang et al's and Zhou et al's, actually our system 2 (i.e. the system only use comparable corpora) has had great superiority than their work (more than 11.5% accuracy). So, compared to our sense disambiguation and pseudo-cue filtering, our web-driven comparable corpora mining provides the main improvement. Actually, our System 2 is partially the same with Zhou et al's work: both predict connectives and perform Explicit-to-Implicit inference. However, Zhou et al's work only use a Tri-gram language model to search consistent expression patterns in small-scale local dataset (PDTB), which results in strict limitation for pattern mining. Especially the patterns (more like phrase pairs) cannot effectively reflect the comparable relation among argument pairs, hardly to support connective predication. By contrast, our web-driven method can obtain abundant candidate comparable resources from large-scale web data, and further our comparable similarity algorithm can simultaneously ensure the similarity in both text spans (respectively in preposition and postposition) of argument pair. Thus our method improves the capacity of comparable corpora mining by performing from-loose-to-tight similarity measurement.

In spite of that, Zhou et al's work, as well as ours, shows better performance than Wang et al's work. This illustrates the feasibility of a statistical model in implicit relation recognition. Especially, with the help of relatively easy relation sense mapping, the cross-argument inference mechanism avoids complicated linguistic analysis, reducing the probability of being misled by mistakes of intermediate steps. Besides, because of the high accuracy (more than 90%, Pitler and Raghupathy, 2008) [2] of explicit relation classification, the Explicit-to-Implicit inference can be jointly used with the explicit classification to generate automatic processing chain of connective identification, sense mapping and relation inference.

However, our best performance is still low with only 6.75% improvement than the baseline. And Wang et al's and Zhou et al's work (Table 11 shows its best performance reported in ACL 2010) are even much worse than the baseline. This reflects implicit relation discourse recognition is very difficult (even for human). Table 11 shows the capacity of human being (two second year postgraduates working on discourse relation recognition) in recognizing the relations, which achieves only 61.29% at best. It is because implicit relation inherently involves subjectivity and ambiguity, e.g. instance (10) shows two possible relations when using different connectives *so* (Contingency) and *but* (Comparison):

(10) He worked all night yesterday, [Implicit= **so** or **but]** slept all day today. (Both Contingency and Comparison are reasonable)

6. CONCLUSION

We have presented the first study on using unsupervised method to recognize implicit discourse relation, as well as the statistical method of relation sense disambiguation. Unlike prior work, our method bypasses the complicated linguistic analysis, and performs Explicit-to-Implicit relation mapping by well using comparable corpora in large-scale web data. Results on PDTB show a significant improvement.

Our analysis revealed several difficulties in current implicit relation recognition: 1) rhetorical structure is important feature for implicit relation analysis, but the rhetoric analysis itself is a hard work so far; 2) discourse relation recognition involves subjectivity and ambiguity, see instance (10), so it shouldn't proceed without the support of context analysis, however, to search the most relative context will bring in another relation analysis; 3) modal connectives (e.i. connectives act as modal particle) normally make pseudo-cues for implicit relation inference, e.g. "I *love* you, *and* [modal particle] I *hate* you too" should be Comparison relation (by "*love*" and "*hate*") but not Expansion signaled by the modal connective "*and*", therefore, to filter modal connectives and instead use latent relation connectives (e.g. *love* & *hate*) to signal relations is very necessary.

In future work, we focus on the automatic method of modal particle identification with the help of sentiment analysis, and on the basis to deeply explore the latent connective mining and application in discourse relation recognition.

7. ACKNOWLEDGMENTS

This research is supported by the National Natural Science Foundation of China (No. 60970056, 60970057, 61003152, 90920004), the Special fund project of the Ministry of Education Doctoral Program (2009321110006, 20103201110021), the Natural Science Foundation of Jiangsu Province (No. BK2011282), the Major Project of College Natural Science Foundation of Jiangsu Province (No. 11KIJ520003) and the Natural Science Foundation of Jiangsu Province, Suzhou City (SYG201030).

8. REFERENCES

[1] Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B. 2008. The Penn Discourse TreeBank 2.0. In Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008), Marrakesh, Morroco.

[2] Pitler, E., Raghupathy, M., Mehta, H., Nenkova, A., Lee, A., and Joshi, A. 2008. Easily identifiable discourse relations. In Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008), Posters, Manchester, UK, 87–90.

[3] Wang, W. T., Su, J., and Tan, C. L. 2010. Kernel Based Discourse Relation Recognition with Temporal. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 11-16 July, Uppsala, Sweden, 710-719.

[4] Pitler, E., Louis, A., and Nenkova, A. 2009. Automatic sense prediction for implicit discourse relations in text. In Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on NLP of the Asian Federation (ACL-IJCNLP 2009), August, Singapore, 683-691.

[5] Lin, Z. H., Kan, M. Y., and Tou, N. G. H. 2009. Recognizing Implicit Discourse Relations in the Penn Discourse Treebank. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009), Singapore, 343-351.

[6] Carlson, L., Marcu, D., and Okurowski, M. E. 2001. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In Proceedings of the Second SIGdial Workshop on Discourse and Dialogue (SIGDIAL2001), 1-2 September, Aalborg, Denmark, 1–10.

[7] Mann, W. C., and Thompson, S. A. 1988. Rhetorical structure theory: Towards a functional theory of text organization. Text, 8(3), 243-281.

[8] Soricut, R., and Marcu, D. 2003. Sentence Level Discourse Parsing Using Syntactic and Lexical Information. In Proceedings of the 2003 Conference of the North America Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL 2003), 27 May - 1 June, Edmonton, Canada, 149-156.

[9] Marcu, D., and Echihabi, A. 2002. An unsupervised approach to recognizing discourse relations. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002), Morristown, NJ, USA, 368–375.

[10] Saito, M., Yamamoto, K., and Sekine, S. 2006. Using phrasal patterns to identify discourse relations. In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2006), June, New York, USA, 133-136.

[11] Wellner, B., Pustejovsky, J., Havasi, C., Rumshisky, A., and Sauri, R. 2006. Classification of discourse coherence relations: An exploratory study using multiple knowledge sources. In Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue, July, Sydney, Australia.

[12] Wolf, F., and Gibson, E. 2005. Representing discourse coherence: a corpus-based analysis. In Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004), Morristown, NJ, USA, 134–140.

[13] Zhou, Z. M., Xu, Y., Niu, Z. Y., Lan, M., Su, J., and Tan, C. L. 2010. Predicting Discourse Connectives for Implicit Discourse Relation Recognition. In Proceedings of the 23th International Conference on Computational Linguistics (COLING 2010), Poster, August, Beijing, China, 1507-1514.

[14] Eleni, M., Robaldo, L., Lee, A., and Joshi, A. 2008. Sense annotation in the penn discourse treebank. Computational Linguistics and Intelligent Text Processing, Lecture Notes in Computer Science, 4919:275-286.

[15] Davidson, I., and Ravi, S. S. 2005. Agglomerative Hierarchical Clustering with Constraints: Theoretical and Empirical Results. In Proceedings of 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), Porto, Portugal, 59-70.

[16] Budanitsky, A., and Hirst, G. 2001. Semantic Distance in WordNet: An experimental, application-oriented evaluation of five neasures. In Proceedings of (NAACL-2000) Workshop on WordNet and Other Lexical Resources, June, Pittsburgh, PA, UAS, 29-34.