Discourse Relation Sense Classification with Two-Step Classifiers

Yusuke Kido
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku
Tokyo, Japan
y.k@is.s.u-tokyo.ac.jp

Akiko Aizawa
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo, Japan
aizawa@nii.ac.jp

Abstract

Discourse Relation Sense Classification is the classification task of assigning a sense to discourse relations, and is a part of the series of tasks in discourse parsing. This paper analyzes the characteristics of the data we work with and describes the system we submitted to the CoNLL-2016 Shared Task. Our system uses two sets of two-step classifiers for Explicit and AltLex relations and Implicit and EntRel relations, respectively. Regardless of the simplicity of the implementation, it achieves competitive performance using minimalistic features.

The submitted version of our system ranked 8th with an overall F_1 score of 0.5188. The evaluation on the test dataset achieved the best performance for Explicit relations with an F_1 score of 0.9022.

1 Introduction

In the CoNLL-2015 Shared Task on Shallow Discourse Parsing (Xue et al., 2015), all the participants adopted some variation of the pipeline architecture proposed by Lin et al. (2014). Among the components of the architecture, the main challenges are the exact argument extraction and Non-Explicit sense classification (Lin et al., 2014).

Argument extraction is a task to identify two argument spans for a given discourse relation. Although the reported scores were relatively low for these components this is partially because of the "quite harsh" evaluation[1]. This led to the introduction of a new evaluation criterion based on partial argument matching in the CoNLL-2016 Shared Task. On the other hand, the sense classification components, which assign a sense to each discourse relation, continue to perform poorly. In particular, Non-Explicit sense classification is a difficult task, and even the best system achieved an F_1 score of only 0.42 given the gold standard argument pairs without error propagation (Wang and Lan, 2015).

In response to this situation, Discourse Relation Sense Classification has become a separate task in the CoNLL-2016 Shared Task (Xue et al., 2016). In this task, participants implement a system that takes gold standard argument pairs and assigns a sense to each of them. To tackle this task, we first analyzed the characteristics of the discourse relation data. We then implemented a classification system based on the analysis. One of the distinctive points of our system is that, compared to existing systems, it uses smaller number of features, which enables the source code to be quite short and clear, and the training time to be fast. The performance is nonetheless competitive, and its potential for improvement is also promising owing to the short program.

This paper aims to reorganize the ideas about what this task actually involves, and to show the future direction for improvement. It is organized as follows: Section 2 presents the data analysis. Then the implementation of the system we submitted is described in Section 3. The experimental results and the conclusion are provided in Section 4 and 5.

2 Data Analysis

There are four types of discourse relations, i.e., Explicit, Implicit, AltLex, and EntRel. In the official scorer, these discourse relations are

[1] CoNLL 2016 Shared Task Official Blog http://conll16st.blogspot.com/2016/04/partial-scoring-and-other-evaluation.html

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 129–135,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

divided into two groups, namely, Explicit and Non-Explicit relations, and they are evaluated separately. AltLex relations are classified into Non-Explicit relations, but they share some characteristics with Explicit relations in that they have words that explicitly serve as connective words in the text. These connective words are one of the most important features in sense classification, as explained later; therefore, we divide the types of relations into (i) Explicit and AltLex and (ii) Implicit and EntRel types in this analysis. Throughout this paper, we do not distinguish between Explicit connective and words that work as connective in AltLex relations, and they are simply referred to as connective.

2.1 Explicit and AltLex Discourse Relations

In the sense classification of Explicit and AltLex relations, connective words serve as important features.

Figure 1 shows the distribution of sense per connective word over the Explicit relations. For example, 91.8% of relations with connective word *and* are labeled as Expansion.Conjunction, and 5.8% as Contingency.Cause.Result. As can be seen, each kind of connective word is mostly covered by only a few senses. Some words such as *also* and *if* have more than 98.8% coverage by a single sense.

According to this observation, it is easy to build a reasonably accurate sense classifier simply by taking connective words as a feature. For example, one obvious method is a majority classifier that assigns the most frequent sense for the relations with the same connective words in the training dataset. Figure 2 shows the accuracy per sense of such a classifier in the training dataset. The method is rather simple, but it achieves more than 80% accuracy for most of the senses.

One exception is Comparison.Concession, which had only a 17.4% accuracy. This is a sense derived from Comparison.Concession and Comparison.Pragmatic concession in the original PDTB, and applies "when the connective indicates that one of the arguments describes a situation A which causes C, while the other asserts (or implies) $\neg C$" (Prasad et al., 2007). Discourse relations with connective words such as *although*, *but*, and *however* are assigned this sense. In the evaluation using the development data, the system assigned Comparison.Contrast to most discourse relations labeled as Comparison.Concession in the golden data. Table 1 shows the senses the system assigned. For example, some of the discourse relations that have a connective word *while* are labeled as Comparison.Concession in the golden data, but the system assigned them as Comparison.Contrast.

Table 1: System output for discourse relations that are labeled as Comparison.Concession in the golden data. The left and right columns show the connective words and the sense assigned by the system, respectively.

Connective	Assigned Sense
while	Comparison.Contrast
even though	Comparison.Concession
still	Comparison.Contrast
nevertheless	Comparison.Contrast
but	Comparison.Contrast
yet	Comparison.Contrast
though	Comparison.Contrast
nonetheless	Comparison.Concession
even if	Contingency.Condition
although	Comparison.Contrast

According to the annotation manual, Contrast and Concession are different in that only Concession has directionality in the interpretation of the arguments. Distinguishing these two senses is, however, ambiguous and difficult, even for human annotators.

2.2 Implicit and EntRel Discourse Relations

By definition, Implicit and EntRel relations have no connective words in the text, which complicates the sense classification task considerably. Other researchers overcame this problem by applying machine-learning techniques such as a Naive Bayes classifier (Wang and Lan, 2015) or AdaBoost (Stepanov et al., 2015). They use various features including those obtained from parses of the argument texts.

As a baseline, we first implemented a support vector machine (SVM) classifier taking a bag-of-words of tokens in the argument texts as features. The evaluation was found to assign EntRel to a large part of the input data. This trend is particularly noticeable for relatively infrequent senses. This problem is partially attributable to the unbalanced data. In fact, there are more EntRel instances included in the training data than

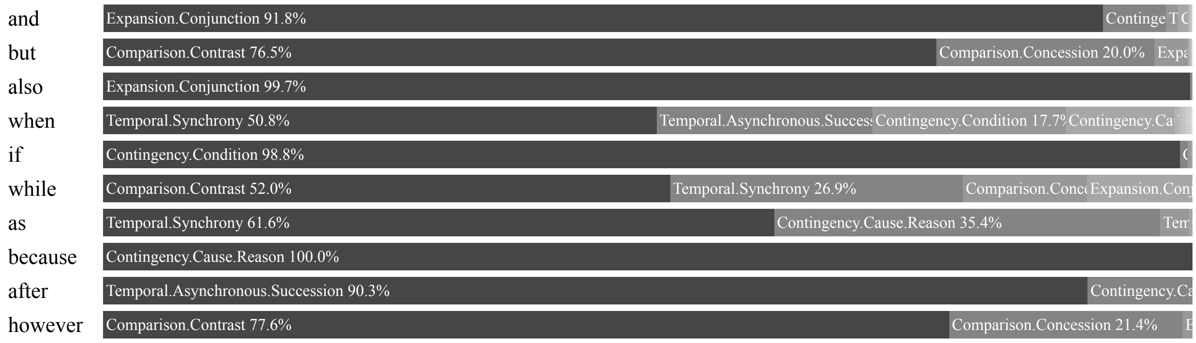

Figure 1: Distribution of the sense assigned to each connective word. All explicit relations with the ten most frequent connective words are extracted from the official training data for the CoNLL-2016 Shared Task.

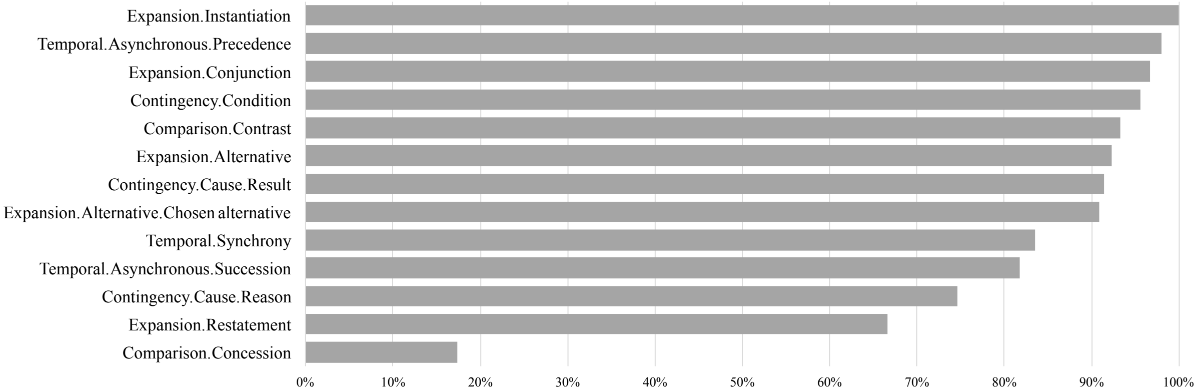

Figure 2: Accuracy of a simple majority classifier that assigns the most popular sense of the discourse relations in the training data with the same connective. Training was conducted on the official training data, and the evaluation used the development data.

the most frequent Implicit sense. We also tried automated weight balancing of the SVM classifier, but the accuracy gain was small.

3 Proposed System

We describe the implementation of our system based on the analysis above. First, the system classifies a discourse relation into two categories, namely (i) Explicit and AltLex or (ii) Implicit and EntRel. This classification is determined simply by checking whether the relation has connective words annotated in the text. The input is then passed to the next two-step classifier components. The following sections detail the three components, i.e., (i) Unknown Connective Substitution (*CS*), (ii) Explicit and AltLex Sense Classifier including Concession vs. Contrast Classifier (*CC*), and (iii) Implicit and EntRel Sense Classifier (*IE*). Figure 3 shows the system overview.

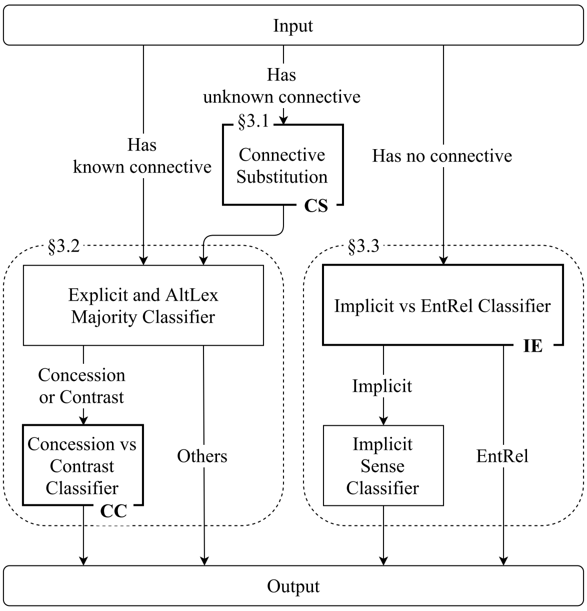

Figure 3: Pipeline of our system.

3.1 Unknown Connective Substitution

If a discourse relation is classified into an Explicit and AltLex category, it will then be passed to a simple majority classifier, i.e., the most frequent sense in the training dataset with the same connective word is assigned. If connective words are alternatively lexicalized, then instances with the same connective words are not necessarily found in the training data. In that case, the majority classifier does not know which sense to assign, whereupon we apply a preprocess, named *unknown connective substitution*, to find a clue for the classifier.

First, the connective words are mapped to a real vector using skip-gram neural word embeddings (Mikolov et al., 2013). For connective words with more than one word, the average vector of every word weighted by term frequency is used. Using this vector, the known connective words in training data that are the closest to the unknown connective words are looked up. Then the connective words are substituted with the closest one, and passed to the next process. Thus, we can use this substitution to reduce the difference between Explicit and AltLex such that it can be ignored, which contributes to the reusability of the components.

3.2 Explicit and AltLex Sense Classifier

As already mentioned, the Explicit and AltLex sense classifier is a majority classifier. It assigns the most popular sense in the training examples that have the same connective words (or those substituted in the pre-process) with the input. Although this classifier already had reasonably good accuracy at this point, we improved it by analyzing which pair of senses are confusing and difficult to distinguish.

In the previous section, we saw that distinguishing between Comparison.Concession and Comparison.Contrast is difficult. The system attempts to solve this problem by repeating the classification using another classifier in cases in which the output of the classifier was Comparison.Concession or Comparison.Contrast. For the second classifier, we use the following features:

1. the connective words,
2. the Arg1 and Arg2 texts: the frequency count of the tokens in the argument texts converted into integer vectors (bag-of-words),
3. the nodes of the parse trees Arg1 and Arg2: similarly to 2, the frequency count of the nodes of the parse trees of argument texts, and
4. the MPQA subjectivity lexicon: each token in the argument texts is classified into nine groups according to the MPQA lexicon, and the number of tokens was counted, ignoring words not in the lexicon.

These are chiefly general-purpose features and widely used in various NLP tasks, and actually a subset of the features used in several previous studies including (Lin et al., 2014) and (Pitler and Nenkova, 2009).

3.3 Implicit and EntRel Sense Classifier

Similar to the Explicit and AltLex sense classification, the Implicit and EntRel sense classification is also a two-step process: first it is determined whether the type is Implicit or EntRel, and then a sense is assigned if classified as Implicit.

Connective words themselves cannot be used as features in Implicit and EntRel sense classification; therefore, other features need to be prepared. There are many candidates for the features. Here, to simplify the implementation, and also because we cannot afford the time for task-specific feature engineering, we merely reuse the same features of the Concession vs. Contrast classifier described in the last section.

4 Experiments

4.1 Experimental Settings

We trained our system on the official training dataset of the CoNLL-2016 Shared Task, and evaluated it on several test datasets. We implemented SVM classifiers, which are popular among various NLP tasks, and MaxEnt classifiers, which have been used in the previous studies. Both are implemented using scikit-learn (Pedregosa et al., 2011), with the default parameters except for the automated weight balancing between classes (`class_weight='balanced'`) in order to overcome the imbalance of the data distribution[2]. In the balanced mode, the weights of samples are automatically adjusted inversely proportional to class frequencies in the input data. We

[2] It should be noted, however, that we also conducted an evaluation on the test and blind test dataset without weight balancing, and found that its effect is small.

Table 2: Experimental results using the two datasets. F_1 scores are shown. "Maj" = majority classifier for Explicit and AltLex relations. "CS" = substitution of unknown AltLex connectives. "IE" = Implicit vs. EntRel classification before Implicit sense classification. "CC" = Concession vs. Contrast classification after Explicit and AltLex sense classification.

	test			**blind-test**		
	All	**Explicit**	**NonExp**	**All**	**Explicit**	**NonExp**
Maj+SVM (Baseline)	0.5116	0.8991	0.1589	0.4404	0.7495	0.1776
Maj+SVM (TIRA Official)	0.5473	0.9022	0.2261	0.5188	0.7543	0.3231
Maj+MaxEnt	0.6093	0.9002	0.3445	0.5215	0.7532	0.3247
Maj+MaxEnt+CS	**0.6145**	**0.9046**	**0.3504**	0.5257	0.7622	0.3241
Maj+MaxEnt+CS+IE	0.5540	**0.9046**	0.2340	0.5290	0.7622	**0.3308**
Maj+MaxEnt+CS +CC	0.5866	0.8460	**0.3504**	0.5357	**0.7838**	0.3241
Maj+MaxEnt+CS+IE+CC	0.5261	0.8460	0.2340	**0.5389**	**0.7838**	**0.3308**

also attempted hyperparameter tuning using the development dataset, but the performance was almost the same.

As a baseline, the majority classifier described in Section 2.1 is used for Explicit and AltLex relations, and an SVM classifier is used for Implicit and EntRel relations. The features for the SVM classifier were bag-of-words of Arg1 and Arg2 texts. The system used in the official evaluation on TIRA was an old version because of deployment problems. This means it is almost the same as the baseline system, except that the MPQA subjectivity lexicon is added as features.

The systems are evaluated using the script provided by the CoNLL-2016 Shared Task organizers. The official evaluation is carried out on TIRA (Potthast et al., 2014).

4.2 Results

Table 2 lists the F_1 scores our systems achieved in the evaluation using the test and blind-test datasets. In the first column, "CS" indicates the substitution of unknown AltLex connectives. "IE" indicates that the Implicit vs EntRel classifier was used, and "CC" indicates the Concession vs. Contrast classifier. A comparison of the two classification algorithms revealed that MaxEnt classifiers were more effective than SVM. This is because SVM is unsuitable for this text classification problem, because text data is high dimensional and sparse. The training of MaxEnt classifiers took only 40 minutes in the longest case, but SVM classifiers required more than 10 hours. In the evaluation using the blind-test dataset, the performance of our system was optimal with the full functions. The blind-test dataset is taken from Wikinews materials; thus, these results imply a good generalization of our system.

4.2.1 AltLex Connective Substitution

As can be seen from the third and fourth columns in Table 2, the substitution of unknown connectives using skip-gram described in Section 3.1 contributed to an improvement on average. Table 4 presents examples of substituted unknown AltLex connectives. The words in the first column are found in AltLex relations, but they are not included in the training data. By applying the substitution preprocess, the known connectives shown in the second column are found to be the closest. As a result, the senses in the third column were chosen by the majority classifier. The fourth column shows the golden sense. This process worked well in the cases of the first three rows. The last two rows are examples of failure. The connective *one reason is that* introduces the following clause as the reason for the preceding phrases, but the word *reason* was omitted from the substituted connective, causing misclassification into Contingency.Cause.Result. In order to distinguish Result and Reason, the system has to consider the word order, but now its information is omitted during the mapping from words to real vectors. In addition, the word2vec model used in this system is a pre-trained model, and it does not include functional words such as *and* or *a*. These words play an important role for our purpose; therefore, an unprocessed model should be used.

4.2.2 Features

We also conducted experiments using different sets of the features. The results are provided in

Table 3: Experimental results using different sets of the features. F_1 scores are shown. Feature 1 = tokens in argument texts. Feature 2 = parse tree nodes of argument texts. Feature 3 = MPQA subjectivity lexicon. All classifiers share these features, and they also use connective words as a feature.

	test			**blind-test**		
	All	**Explicit**	**NonExp**	**All**	**Explicit**	**NonExp**
Features 2+3	0.5717	**0.9067**	0.2661	0.4883	0.7604	0.2571
Features 1 +3	0.6036	0.9056	0.3287	0.4950	0.7617	0.2674
Features 1+2	**0.6160**	0.9035	**0.3544**	0.5228	0.7617	0.3195
Features 1+2+3	0.6145	0.9046	0.3504	**0.5257**	**0.7622**	**0.3241**

Table 4: Preprocessing results on AltLex relations with unknown connective words.

Unknown Connective	**Closest Connective**	**Output**	**Golden Sense**
the delay resulted from	the rise resulted from	**Contingency.Cause.Reason**	Contingency.Cause.Reason
that change will obviously impact	that will cinch	**Contingency.Cause.Result**	Contingency.Cause.Result
that rise came on top of	on top of that	**Expansion.Conjunction**	Expansion.Conjunction
one reason is that	that is why	Contingency.Cause.Result	Contingency.Cause.Reason
one reason is	one is	Expansion.Instantiation	Contingency.Cause.Reason

Table 3. The score is lowest when the token feature is omitted, except for the Explicit relations in the test dataset. The impact of the MPQA feature is small but not expectable, which led to the unstable results.

5 Conclusion

We analyzed the characteristics of the data used in the CoNLL-2016 Shared Task and described the implementation details of our system. The performance on the Implicit and EntRel sense classification task is still low and has room for improvement. These results imply that these tasks are essentially difficult and require a deeper understanding of semantics, pragmatics, and background knowledge behind the text. A more detailed analysis of the materials is essential to effectively improve the performance on these tasks.

6 Acknowledgments

This work was supported by the Japan Society for the Promotion of Science KAKENHI Grant Number 15H02754 and 16K12546.

References

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Emily Pitler and Ani Nenkova. 2009. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ACLShort '09, pages 13–16, Stroudsburg, PA, USA. Association for Computational Linguistics.

Martin Potthast, Tim Gollub, Francisco Rangel, Paolo Rosso, Efstathios Stamatatos, and Benno Stein. 2014. Improving the reproducibility of PAN's shared tasks: Plagiarism detection, author identification, and author profiling. In Evangelos Kanoulas, Mihai Lupu, Paul Clough, Mark Sanderson, Mark Hall, Allan Hanbury, and Elaine Toms, editors, *Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 5th International Conference of the CLEF Initiative (CLEF 14)*, pages 268–299, Berlin Heidelberg New York, September. Springer.

Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. 2007. The Penn Discourse Treebank 2.0 annotation manual.

Evgeny A. Stepanov, Giuseppe Riccardi, and Ali Orkan Bayer. 2015. The UniTN discourse parser in CoNLL 2015 Shared Task: Token-level sequence labeling with argument-specific models. In *Proceed-*

ings of the 19th Conference on Computational Natural Language Learning: Shared Task, pages 25–31.

Jianxiang Wang and Man Lan. 2015. A refined end-to-end discourse parser. In *Proceedings of the 19th Conference on Computational Natural Language Learning: Shared Task*, pages 17–24.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Rashmi Prasad, Christopher Bryant, and Attapol Rutherford. 2015. The CoNLL-2015 shared task on shallow discourse parsing. In *Proceedings of the 19th Conference on Computational Natural Language Learning: Shared Task*, pages 1–16.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Bonnie Webber, Attapol Rutherford, Chuan Wang, and Hongmin Wang. 2016. The CoNLL-2016 shared task on multilingual shallow discourse parsing. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning - Shared Task*, Berlin, Germany, August. Association for Computational Linguistics.