Discourse Relations and Conjoined VPs: Automated Sense Recognition

Valentina Pyatkin
Department of Computer Science
Sapienza University of Rome
pyatkin@di.uniroma1.it

Bonnie Webber
School of Informatics
University of Edinburgh
bonnie@inf.ed.ac.uk

Abstract

Sense classification of discourse relations is a sub-task of shallow discourse parsing. Discourse relations can occur both across sentences (*inter-sentential*) and within sentences (*intra-sentential*), and more than one discourse relation can hold between the same units. Using a newly available corpus of discourse-annotated intra-sentential conjoined verb phrases, we demonstrate a sequential classification system for their multi-label sense classification. We assess the importance of each feature used in the classification, the feature scope, and what is lost in moving from gold standard manual parses to the output of an off-the-shelf parser.

1 Introduction

Discourse relations can hold between inter-sentential and intra-sentential arguments. As Language Technology has much to gain from recognizing intra-sentential discourse relations (Joty et al., 2015), the Penn Discourse TreeBank project has annotated the discourse senses of conjoined verb phrases in the Wall Street Journal corpus (Webber et al., 2016).

Broadly construed, conjoined VPs are sisters in a parse tree, separated from each other by a conjunction and/or punctuation, and possibly one or more adverbs or adverbial phrases as well. As with other units of discourse, more than one sense relation can hold between conjoined VPs. An explicit conjunction may itself convey multiple senses, or additional senses may arise through inference or be signaled with other lexico-syntactic cues (Webber et al., 2016; Prasad et al., 2014). With no explicit conjunction, sense relations will arise through inference or are signaled with other lexico-syntactic cues. Example (1) illustrates senses arising through inference, even though an explicit connective is also found in the conjunction. Here, 'making the penalties fairer and easier to administer' is the GOAL of 'simplifying the penalties', and the latter is the MANNER of achieving that goal.

(1) Long-debated proposals to *simplify the more than 150 civil penalties* (ARG1) *and make them fairer and easier to administer*(ARG2) are in the House tax bill. [wsj_0293]

Automatic classification of the sense relations that hold between sister VPs can thus be formulated as the following task: given a pair of sister VPs and how they have been conjoined, can the sense relation(s) between them be induced? We have approached this task using two Support Vector Machines in a way that allows multi-label classification. To understand what is contributing to effective classification, we examine the separate contributions of syntactic (Section 4.3) and semantic features (Section 4.4), and then the extent to which information internal to the sister VPs suffices to determine how they relate to one another, or whether features external to the pair are also needed (Section 4.5). We also assess the extent to which performance drops when argument spans are provided by an 'off-the-shelf' parser rather than manual annotation (Section 5).

The novel contribution of this work is its use of multi-label classification in determining the discourse sense(s) that hold between conjoined VPs. This type of sense classification on conjoined VPs has not been done before to our knowledge. The evaluation of the features and the feature scope could provide a useful starting-point for future systems that classify inter-sentential discourse relations. Such a classifier could be in-

Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics, pages 33–42, Valencia, Spain, April 3-7 2017.

corporated into other NLP systems, such as Machine Translation or Automatic Summarization. Louis et al. (2010), for example, showed the benefit of discourse features as importance indicators for automatic summarization, Meyer et al. (2015) used sense labeled discourse connectives in an improved phrase based machine translation system and Prasad and Joshi (2008) generated questions using properties and arguments of specific discourse relations.

2 Background

The sense annotation of discourse relations is part of shallow discourse parsing, involving the identification of pairs of discourse arguments (*Arg1* and *Arg2*) and the sense(s) i which they are related.

(2) Exxon Corp. *built the plant* (ARG1) **but** *closed it in 1985* (ARG2). [wsj_1748]

Example (2) shows the two arguments and the explicit connective 'but'. The annotators labeled this as expressing both CONCESSION (i.e., closing was not expected) and PRECEDENCE (i.e., closing occurred after building). Discourse relations are signaled either explicitly through a discourse connective, or implicitly, or with some other lexicalization (ALTLEX) such as 'will result in'. In the conjoined VP sub-corpus of the PDTB 3.0 (Webber et al., 2016), the left argument is labeled *Arg1* and the right argument, *Arg2*. The goal of shallow discourse parsing is thus to automatically identify the arguments, their spans, the connective (for an explicit relation), and the sense(s) in which they are related. It is called 'shallow' because it does not recursively construct a discourse 'parse tree' (Stede, 2011). The first end-to-end shallow discourse parsers carried out subtasks in a pipeline, separating the tasks of parsing explicit and implicit discourse relations (Lin et al., 2014; Wang and Lan, 2015).

Shallow discourse parsing of conjoined VPs differs from this model of discourse parsing in that the arguments must be sister VPs in a parse tree. Thus, syntactic parsing (either phrase-structure or dependency) must precede identification of sister VPs, whether there is an explicit connective between them or not. This makes shallow discourse parsing more dependent on parser accuracy than in the past. As we will show in Section 5, parsers often fail to accurately parse conjoined VPs (or conjoined structures in general, (Ficler and Goldberg, 2016)).

In terms of features, Subba and Di Eugenio (2009) mention VerbNet as a resource to generalize the semantics of verbs. Pitler and Nenkova (2009) used a small collection of syntactic features to do single-label sense classification from a set of four high-level sense types. Rutherford and Xue (2014) mention that Brown Clusters are helpful to classify implicit relations. For the machine learning algorithms, Meyer et al. (2015) claim that a Maximum Entropy classifier is suitable for sense classification as it learns feature combinations. Hernault et al. (2010) propose the use of a SVM for its suitability for a larger feature-space.

3 Corpus

The Penn Discourse TreeBank has been extended to cover discourse relations between conjoined VPs occurring in the Penn Wall Street Journal corpus (Webber et al., 2016). Besides this sub-corpus, we are aware of only one corpus of discourse annotated conjoined VPs (Subba and Di Eugenio, 2009). This contains fewer annotated tokens than the current set ($\sim$600, as opposed to $\sim$4600), with several sense labels specific to the instruction domain and with only a single relation able to hold between any two conjuncts.

A total of 4633 conjoined VPs have now been annotated in the PDTB, with 3372 having a single sense and 1261 having multiple senses (Webber et al., 2016). There are three conditions in which multiple sense relations hold between sister VPs[1]:

1. Two Explicit senses: One sense is associated with the explicit conjunction and another with an explicit adverb (e.g. "and later").

2. Explicit and Implicit senses: One sense is associated with the explicit conjunction, while other senses are derived through inference.

3. Explicit and AltLex senses: One sense is associated with the explicit conjunction, while another is expressed through an AltLex (e.g. "at the same time").

The numbers for the three types of multi-label conjunctions can be seen in Table 1, along with the numbers for single-label conjunctions. If there is no explicit connective, the multi-sense relations

[1]There could also have been multiple implicit relations between sister VPs, but none appear in the Conjoined VP subcorpus.

are annotated on a single instance of the conjunction. In cases where one sense comes from the explicit conjunction, while the others are derived through inference, this is implemented as two separate linked tokens, one labeled "Explicit", the other "Implicit". This means that some implicit relations hold between sister VPs with no explicit conjunction between them, and others hold between explicitly-conjoined sister VPs whose additional senses derive through inference. A revised

	single-s.	**multi-s.**
Explicit conjunction	2933	
Explicit adverbial	29	
Implicit (punctuation)	410	
Explicit + Adverbial		214
Explicit + Implicit		1017
Explicit + AltLex		30

Table 1: Single-sense and multi-sense counts.

set of sense labels, consisting of 34 labels, has been used in annotating the Conjoined VP corpus and other recent annotation of the Penn Discourse TreeBank (Webber et al., 2016). The senses of the PDTB are constructed in a hierarchical manner. The first level of the hierarchy distinguishes between 4 different sense categories: TEMPORAL, COMPARISON, CONTINGENCY and EXPANSION (Prasad et al., 2014).

4 Classification

4.1 Baseline

As there currently exists no sense-relation classification system for conjoined VPs, the strongest baseline corresponds to majority properties of the corpus. Different majority classes are attributed to implicit and explicit conjunction. For explicit conjunctions with a connective/adverb, the most common sense per connective/adverb is chosen. For implicit relations the most common implicit sense is selected (TEMPORAL.ASYNCHRONOUS.PRECEDENCE). We apply these rules on the same dataset that is used for the classification approach, with certain senses removed, as will be explained in Section 4.2. The various baselines can be seen in Table 2

4.2 Classification approach

Since several senses occur only rarely in the corpus, while EXPANSION.CONJUNCTION occurs as a sense label on more than 77 % of the tokens, actions had to be taken to avoid optimizing performance by simply learning the majority label. To avoid this false optimization, we only considered senses that occurred at least 30 times in the corpus, and in any given training set, we only allowed up to 500 tokens of EXPANSION.CONJUNCTION. The final sense set used for classification thus consists of the 11 senses in Table 3. Tokens not annotated with at least one of these senses have been removed, and multi-label tokens with only one sense shown in Table 3 have been included as single-label tokens. As a result 2446 conjunctions can be used for training and testing.

A system with two classifiers is used for the multi-label classification task. To prove the effectiveness of this approach, in Section 6 we compare this two-classifier method with another multi-label

	Acc.	Prec.	Rec.	F-m.
Implicit	0.37	0.14	0.37	0.20
Explicit	0.49	0.61	0.49	0.42
Total	0.49	0.58	0.49	0.41

Table 2: Baseline for only implicit relations, only explicit relations and the total dataset.

Comparison	Concession	Arg2-as-denier
Comparison	Contrast	
Contingency	Cause	Result
Contingency	Purpose	Arg2-as-goal
Expansion	Conjunction	
Expansion	Disjunction	
Expansion	Level-of-detail	Arg2-as-detail
Expansion	Manner	Arg1-as-manner
Expansion	Substitution	Arg1-as-subst
Expansion	Substitution	Arg2-as-subst
Temporal	Asynchronous	Precedence

Table 3: The subset of 11 senses used in our classification. The left-hand column shows the high-level category of the relation, and the center column shows mid-level sense category. For senses in which a relation can hold in either direction, the right-hand column specifies which direction holds. In the case of Substitution, both the sense in which Arg1 serves as a substitute for Arg2 (i.e., Arg1-as-subst) and the sense in which Arg2 serves as a substitute for Arg1 (i.e., Arg2-as-subst) are used in classification.

classification approach using a One-Vs-Rest classifier, which employs a separate SVM for every label (Pedregosa et al., 2011). The classification setup can be seen in Figure 1. Two SVM classifiers are trained and tested independently. One classifier, called 'Implicit' classifier, is trained on instances of implicit conjunctions and conjunctions with alternative lexicalizations or discourse adverbials. The 'Explicit' classifier is trained on instances of explicit conjunctions. While relations arising from AltLex or discourse adverbials could technically be seen as explicit conjunctions, we added them to the 'Implicit' classifier's training set for the system to be able to identify multi-label conjunctions containing both an explicit connective and an adverbial/AltLex. As part of the training data, both classifiers are also given negative instances, e.g. training data from the respective other classifier, which the classifier ideally has to label as 'NO'.

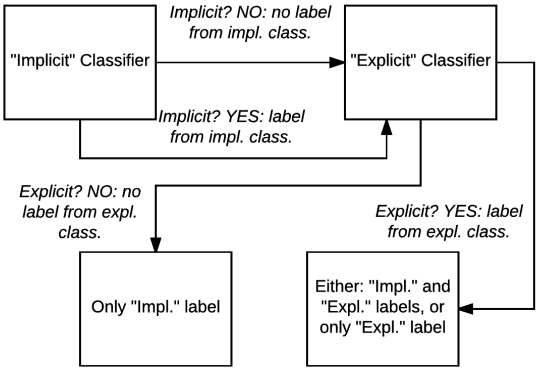

Figure 1: Classification system using two classifiers and negative examples (the order of the two classifiers does not matter as they are independent).

The system starts with both classifiers running in parallel on the same instance. This instance is then assigned an (implicit) sense or is classified as a non-implicit relation by the Implicit classifier and either assigned an (explicit) sense or classified as a non-explicit relation by the Explicit classifier. The order in which the two classifiers are applied is arbitrary, since they operate independently of each other.

After both classifiers finish, their results are combined. The set of the labels from both classifiers, with the NO labels removed, is then the final multi- or single-label instance. This allows for single-label classification, as well as the multi-label cases mentioned in Section 3. A drawback of making both classifiers also predict 'NO' labels is that it could result in both classifiers predicting 'NO', indicating that the system cannot associate any relation to that instance.

As both classifiers learn their parameters independent of the other classifier, the feature selection and evaluation is kept separate for each classifier. The performance of the classifiers is reported using precision, recall and f1-measure. All three measures are calculated for each class separately and then averaged. The f1-measure is also weighted by the number of class-instances, which results in numbers that do not lie between the recall and the precision. The feature analysis is done using a Recursive Feature Elimination algorithm (Pedregosa et al., 2011), which designates weights to the individual features by recursively removing features. For the single-label 'Implicit' and 'Explicit' classifiers the reported measures are obtained using 4-fold cross-validation.

4.3 Syntactic Features

4.3.1 Experiments

Since the connective and its sense-dependent distribution are used in the baseline, each possible connective is encoded as a binary feature, together with its PoS. Unsurprisingly, the use of only this feature results in a better accuracy for the 'Explicit' classifier (0.54 +/- 0.01) than the 'Implicit' classifier (0.50 +/- 0.04). As noted earlier, implicit sense relations can occur along with explicit conjunctions, when these relations are taken to be inferred from the arguments (and possibly their context), rather than being linked to the explicit conjunction. This property explains why the performance of the 'Implicit' classifier is not much worse: while the connective is not signaling the sense explicitly, the classifier can learn that some implicit senses co-occur with certain explicit connectives/senses. Since discourse adverbials such as 'instead' or 'moreover' can explicitly signal discourse relations, they are also added to the feature set, resulting in a slight increase of accuracy and f-measure for the 'Explicit' classifier (0.56 +/- 0.03 and 0.51 +/- 0.04).

Using PoS tags from the PTB corpus, unigram, bigram and trigram PoS features are implemented. The use of ngrams with $n > 1$

is meant to serve as a proxy for syntactic patterns. The PoS features are also weighted using tfIdf. A single ngram functions as the *term*, an instance of the two arguments of a conjunction represents the *document* (we count how many times a certain ngram occurs in the arguments) and the *inverse document frequency* is calculated using all the training instances. Other properties encoded as features include whether or not a comparative or superlative adjective is present in either arguments and whether there is a modal verb. Negation could serve as a useful feature to identify EXPANSION.DISJUNCTION or COMPARISON.CONTRAST (see example (3)).

(3) *...is now willing to pay higher bank fees and interest,* (ARG1) **but** *isn't likely to boost its $965 million equity contribution* (ARG2). [wsj_2172]

A negation feature has been implemented in its simplest form, checking for the 'un-' affix and for certain predefined negation terms such as 'not'. The negation features also specify in which argument the feature was found.

4.3.2 Results

The connective/adverb features are included in all of the experiments. Table 4 displays all of the results. While the syntactic features only increase the recall of the 'Explicit' classifier, the performance of the 'Implicit' classifier is considerably improved when using the PoS tags of the arguments. The contribution of negation can be seen by comparing rows 7 and 8 in Table 3. For explicit relations, negation improves recall while maintaining precision, while for implicit relations, negation decreases recall while improving precision. The improvement comes from a better detection of the sense EXPANSION.LEVEL-OF-DETAIL.ARG2-AS-DETAIL.

For PoS-trigrams, the Recursive Feature Elimination algorithm shows that for both the 'Implicit' and 'Explicit' classifiers, the twenty highest ranked trigram features all include a CC (coordinating conjunction). This is not surprising because (as noted in Section 3) when an explicitly-conjoined VP has additional inferred senses, the convention is to include the conjunction as part of *Arg2*. The most prominent patterns are either CC followed by either CD (cardinal number) or DT (determiner). The Explicit classifier also includes among its highest ranked PoS-trigrams, three that start with CC and IN (preposition or subordinating conjunction) as in Ex. (4), which reflects a deviation from the typical syntax of conjoined VPs, in which a verb follows the conjunction. This standard pattern appears 1015 times in the corpus.

(4) ... fees they can charge *have plunged to almost nothing* (ARG1) **and** *in some cases are just that* (ARG2). [wsj_1600]

It is interesting to see which of the senses are more easily detected with the inclusion of syntactic features. The 'Implicit' classifier, with its most useful feature-setting of 'trigram PoS-tags', improves on all senses except EXP.SUBST.ARG2-AS-SUBST.. The CC, IN construct mentioned earlier appears mainly in implicit CONTINGENCY.CAUSE.RESULT conjunctions. This sense is also the sense whose f-measure improves the most with the inclusion of syntactic features, from 0.58 to 0.70. The cardinal number feature improves the classification of TEMPORAL.ASYNCHRONOUS.PRECEDENCE relations, where the event specified in Arg1 that precedes that specified in Arg2. A total of 84 implicit tokens contain a cardinal number, many of which describe the movement of stock prices over time. (This is a likely consequence of the content of the WSJ corpus.) An example where the explicit sense is EXPANSION.CONJUNCTION and the implicit sense is TEMPORAL.ASYNCHRONOUS.PRECEDENCE, is:

(5) ... Delta *issued 2.5 million shares of common stock to Swissair* **and** *repurchased 1.1 million shares for use in a company employee stock ownership plan.* [wsj_1011]

The 'Explicit' classifier improves only in recall with the addition of syntactic features. Because the tfIdf weighted unigrams of words and PoS work slightly better than the PoS trigrams, one could conclude that single words or PoS are as much indicative of the sense as syntactic combinations of PoS. A reason for this could be that there is not much syntactic variability in the way a VP constituent can be constructed. COMPARISON.CONTRAST and CONTINGENCY.PURPOSE.ARG2-AS-GOAL get recognized, whereas before they were not, but the f-measure of other senses sinks. There is therefore a trade-off between the classification of more senses and the precision of the individual senses

	Explicit				Implicit			
	Accuracy	Prec.	Rec.	f-measure	Accuracy	Prec,	Rec.	f-measure
2g PoS	0.74 (0.09)	0.74	0.72	0.69 (0.10)	0.60 (0.03)	0.60	0.56	0.55 (0.03)
3g PoS	0.74 (0.07)	0.74	0.72	0.69 (0.09)	0.60 (0.07)	0.60	0.59	0.56 (0.06)
1g words+PoS	0.75 (0.07)	0.75	0.73	0.71 (0.08)	0.60 (0.02)	0.60	0.58	0.55 (0.02)
2g words+PoS	0.73 (0.06)	0.73	0.72	0.68 (0.07)	0.60 (0.03)	0.60	0.59	0.55 (0.04)
2g words	0.73 (0.06)	0.73	0.73	0.67 (0.07)	0.59 (0.09)	0.59	0.56	0.54 (0.08)
3g PoS + 1g words	0.75 (0.09)	0.75	0.73	0.70 (0.10)	0.57 (0.03)	0.57	0.55	0.53 (0.02)
synt. feat. no neg.	0.74 (0.09)	0.74	0.67	0.67 (0.11)	0.45 (0.02)	0.45	0.55	0.37 (0.04)
synt. feat. + neg.	0.75 (0.10)	0.75	0.70	0.68 (0.12)	0.51 (0.02)	0.51	0.50	0.45 (0.03)
3g PoS + synt. feat.	0.74 (0.10)	0.72	0.68	0.68 (0.10)	0.54 (0.04)	0.54	0.52	0.51 (0.04)
Conn/Adv	0.75 (0.07)	0.75	0.68	0.67 (0.09)	0.46 (0.02)	0.46	0.52	0.39 (0.03)

Table 4: Comparison of performance of syntactic features. The number in parenthesis is the confidence interval of the cross-validation score. ('1g' stands for unigram, '2g' for bigram etc., 'synt. feat.' stands for comparative/superlative adjectives and modal verbs)

classified, when using syntactic features for the 'Explicit' classifier.

4.4 Semantic Features

4.4.1 Experiments

In order to exploit the semantic content of the conjunctions, multiple semantic resources are used. These resources generally are semantic representation techniques that are able to reduce the dimensionality of the data. Since the task consists of classifying sense relations between two arguments, a representation of the semantic combination of the two arguments might be suitable. For this purpose the Cartesian product between the corresponding representation of the words in *Arg1* and in *Arg2* is constructed.

VerbNet (Schuler, 2005) features are implemented as the Cartesian product of the verbs in the VPs and also as a tfIdf weighted bag-of-words representation. Since we are working with VP conjunctions the role of the verbs is assumed to be important for the sense of the relation.

BrownCluster classes represent words as semantic clusters, through a hierarchical clustering approach using mutual information (Turian et al., 2010). For the BC features the Brown Clusters from the CoNNL-2016 Shared Task[2], containing 100 clusters, are used. Previous research on discourse relations showed that Brown Clusters are especially useful for the classification of implicit relations (Rutherford and Xue, 2014). BC pairs with a hyponym-meronym relation have been shown to be predictive for the EXPANSION sense (Rutherford and Xue, 2014). Again both the Cartesian product and the bag-of-word representation are implemented.

We used WordNet (Miller, 1992) to analyze the semantic relations and similarity of the words between the two arguments. For this purpose the antonymy, synonymy and hypernymy annotations of WordNet are considered. Every noun and verb in the feature scope is assigned to its disambiguated synset, using Banerjee and Pedersen (2002)'s approach of applying the Lesk algorithm to WordNet. The relational features, such as antonymy, are represented as categorical features containing the respective synset. Similarity between the arguments is encoded into a feature by calculating the normalized shortest-path scores between all the synsets of the two arguments.

4.4.2 Results

The three semantic feature-types, BrownCluster, VerbNet and WordNet, are evaluated in combination with the connectives/discourse adverbials features. Table 5 shows that the 'Implicit' classifier profits the most from the semantic features. This indicates that the semantic information contained in a connective, can, to some extent, be found in in the arguments of implicit relations. For explicit relations, the sense of the relation might not have to be expressed semantically in the arguments. In terms of semantic resources, the TfIdf weighted BC features result in the biggest accuracy and f-measure for the 'Implicit' classifier. The 'Ex-

[2] http://www.cs.brandeis.edu/~clp/conll16st/dataset.html

plicit' classifier shows a minimal improvement in f-measure when adding semantic features. The WordNet features seem to be the least indicative for the 'Implicit' classifier, but still offer an improvement compared to the basic feature set.

The Recursive Feature Elimination shows that most of the highly ranked VerbNet classes contain one or more classes that semantically indicate a verbum dicendi, such as 'approve', 'manner_speaking' or 'indicate'. These verbs seem very indicative of the COMP.CONCESSION.ARG2-AS-DENIER sense, as the denying tends to be expressed in the form of reported speech. The highest ranked BC classes are not as easily analyzed, since the clusters do not have names. Nevertheless, clusters with distinct properties can be identified. One highly ranked cluster contains a lot of hyphen separated adjectives, such as 'double-masted', 'ski-masked' and 'well-built'. Most of the instances in the corpus containing such adjectives display one of the EXPANSION senses, where the adjectives are found in ARG2. Another, more semantically motivated cluster, contains company names such as 'Rossignol' and 'Icelandair', which is probably influenced by the financial domain of the corpus.

4.5 Internal and External Features

In the following, features derived from the arguments and connective are considered *internal features*, while features obtained from outside their scope are considered *external features*. The motivation behind this feature scope exploration comes from the distinction between the senses COMPARISON.CONCESSION and COMPARISON.CONTRAST. While both involve a comparison between *Arg1* and *Arg2*, COMPARISON.CONCESSION is used when one expresses an expected situation which is refuted by the other (either ARG1-AS-DENIER or ARG2-AS-DENIER). The implication of an expectation of a situation might require more textual context or even world-knowledge. Both senses exhibit a similar distribution of connectives (*but* and *implicit connective*), making their distinction even harder. To test whether the internal feature scope is enough or whether some external features could contribute to a better sense classification, a combination of syntactic and semantic features is used on the internal, external and combined feature-scope. The results in Table 6 indicate that the arguments contain all of the information needed to classify the sense of conjoined VPs. Adding the external features on top of the internal features results in about the same performance for the 'Explicit' classifier and in a worse performance for the 'Implicit' classifier. The external features seem to mainly add noise to the feature space. The external scope alone results in the worst 'Explicit' classifier performance until now and stays about the same as the connective/adverb features performance of the 'Implicit' classifier. This experiment therefore showed that for the classification of conjoined VPs the most relevant information is contained in the arguments. At the same time, the assumption that features from the external feature scope are useful to distinguish COMPARISON.CONCESSION and COMPARISON.CONTRAST, has been confirmed. Their classification performance is better when using only external features than when using only internal features (see Table 7). This property could, in future work, be used when a separate classifier is built for every sense.

5 Comparison with off-the-shelf parses

The comparison of feature scope goes hand in hand with the comparison of the classifier's performance on gold-standard data versus automatic parses. While the experiments above have used argument spans provided in the annotated corpus, any practical system will have to rely on whatever conjoined VPs have been identified by its parser. When given a sentence containing a conjoined VP, a parser should produce a parse that includes a VP parent, with VP siblings and a connective or comma in between. While the Stanford Shift-reduce Constituency Parser[3] fulfills this condition, it failed to produce a conjoined VP analysis for 1369 of the 4633 tokens in the corpus. Where it did produce an analysis, the analysis often differed from that in the conjoined VP corpus because of the annotation guidelines. For example, the guidelines indicate that parenthetical and non-restrictive relative clauses (as in Ex. (6)) can be omitted if they don't contribute to the sense relation(s) that hold between the conjuncts (Webber et al., 2016). Reported speech and attribution relations also belong to this category.

(6) It is also *pulling 20 people out of Puerto Rico*, who were helping Hurricane Hugo victims, **and** *sending*

[3] http://nlp.stanford.edu/software/ srparser.shtml

	Explicit				Implicit			
	Accuracy	Prec.	Recall	F-m.	Accuracy	Precision	Recall	F-m.
VN TfIdf	0.72 (0.08)	0.72	0.71	0.69 (0.08)	0.51 (0.02)	0.51	0.47	0.48 (0.02)
VN c.p.	0.73 (0.08)	0.73	0.71	0.69 (0.09)	0.53 (0.05)	0.53	0.51	0.49 (0.06)
BC TfIdf	0.72 (0.09)	0.72	0.71	0.69 (0.10)	0.60 (0.06)	0.60	0.58	0.58 (0.05)
BC c.p.	0.73 (0.06)	0.73	0.71	0.69 (0.07)	0.52 (0.06)	0.52	0.51	0.49 (0.08)
WN	0.74 (0.08)	0.74	0.68	0.67 (0.10)	0.48 (0.02)	0.48	0.45	0.44 (0.02)
Conn/Adv	0.75 (0.07)	0.75	0.68	0.67 (0.09)	0.46 (0.02)	0.46	0.52	0.39 (0.03)

Table 5: Comparison of performance of semantic features (BC = BrownCluster, VN = VerbNet., WN = WordNet, c.p. = Cartesian Product, TfIdf = weighted with TfIdf). For comparison the performance using the basic Conn/Adv features is added.

	Explicit				Implicit			
	Accuracy	Prec.	Recall	F-m.	Accuracy	Precision	Recall	F-m.
Internal	0.73 (0.08)	0.73	0.70	0.69 (0.07)	0. 58 (0.02)	0.58	0.56	0.55 (0.02)
External	0.71 (0.07)	0.71	0.67	0.67 (0.08)	0.47 (0.05)	0.47	0.44	0.44 (0.05)
Int. + Ext.	0.73 (0.08)	0.73	0.70	0.70 (0.10)	0.56 (0.04)	0.56	0.53	0.53 (0.04)

Table 6: Comparison of the two classifiers' performance on features from the internal, external and combined feature scope. For comparison the performance using the basic Conn/Adv features is added.

	Ext.	Int.	Int.+Ext.
Comp.Concess.	0.82	0.81	0.79
Comp.Contrast	0.10	0.04	0.19

Table 7: F-m. for COMPARISON.CONCESSION and COMPARISON.CONTRAST given different feature scopes (using the 'Explicit' classifier).

them to San Francisco instead. [wsj_1899]

Another guideline is that the arguments should follow a parallel structure, where words whose scope encompasses both arguments are not included. This most commonly affects adverbs located in front of *Arg1*.

We carried out two experiments with the annotated VPs and the automated parses – the first simply testing on automated parses and the other, both training and testing on the automated parses. The results from Table 8 show that the performance of a classifier decreases in both experiments. The changes in span and the inclusion/exclusion of adverbs has the biggest effect on recall. This emphasizes the importance of the argument spans for sense classification. The worse performance of the training and testing on the parsed data can also be attributed to the smaller amount of training data available.

Train/Test	precision	recall	f1-m.
goldst/goldst	0.62	0.65	0.60
goldst/parses	0.53	0.43	0.46
parses/parses	0.44	0.45	0.43

Table 8: Results of the goldstandard and automatic parses experiments. Only the tokens containing a conjoined VP analysis in the automatic parses were used for these experiments.

6 Discussion of the full system

In this section the whole two-classifier system, with negative training examples, is evaluated and discussed. The 'Explicit' classifier's performance using the connective/adverb as features could only minimally be improved using tfIdf weighted unigram features of both PoS and words. For the final system this classifier uses only these features. The 'Implicit' classifier uses the tfIdf weighted PoS trigrams and the tfIdf weighted Brown Cluster classes. The full system achieves a precision of 0.66, a recall of 0.64 and an f-measure of 0.59. The customized featureset strategy might not be necessary, as using the same featureset for both classifiers also results in an f-measure of 0.61. To motivate the use of the two-classifier system, we compared it to the performance of a One-Vs-

Rest classifier approach (Pedregosa et al., 2011), where a separate SVM classifier is trained for each sense. The O.Vs.R strategy achieves a precision of 0.74, a recall of 0.52 and f-measure of 0.57. While the precision is higher, the recall and (sense-)weighted f-measure is lower. The advantage of the One-Vs-Rest classifier strategy is a higher accuracy of correctly classified multi-label instances (0.42), whereas the system only classifies 30%. The system is better at classifying individual explicit/implicit senses rather than finding multi-sense combinations. Adding negative instances to the classifiers in order to make them predict whether or not an implicit or explicit sense holds is effective. Many of the correctly predicted senses arise from single-label conjunctions, e.g. the system manages to correctly make the classifiers say when either no explicit or no implicit relation holds. The performance of the system is better than the predefined baseline in Table 2. The f-measure increases from 0.41 to 0.59. The baselines of the individual classifiers, e.g. the 'Implicit' and 'Explicit' classifier, have also been beat. The 'Explicit' classifier, with an accuracy of 0.75 and an f-measure of 0.71 is much better than the baseline of 0.49 and 0.42. The 'Implicit' classifier's baseline improves the most, from an accuracy of 0.37 to 0.6 and an f-measure of 0.2 to 0.56. This is not surprising as we only chose one majority class for all of the implicit instances.

7 Conclusion and Future Work

This paper presents the first work on automatic sense-classification of conjoined VPs and hopefully inspires more research on this topic, further improving the classification performance. Since sense labelling is only a subtask of shallow discourse parsing, future work could be concerned with the construction of a complete discourse parser for conjoined VPs. An improved argument detection system could allow a better characterization of the extent to which errors in argument span make a difference in sense classification.

References

Satanjeev Banerjee and Ted Pedersen. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 136–145. Springer.

Jessica Ficler and Yoav Goldberg. 2016. Improved parsing for argument-clusters coordination. In *The 54th Annual Meeting of the Association for Computational Linguistics*, page 72.

Hugo Hernault, Helmut Prendinger, David A DuVerle, Mitsuru Ishizuka, and Tim Paek. 2010. Hilda: a discourse parser using support vector machine classification. *Dialogue and Discourse*, 1(3):1–33.

Shafiq Joty, Giuseppe Carenini, and T. Raymond Ng. 2015. Codra: A novel discriminative framework for rhetorical analysis. *Computational Linguistics, Volume 41, Issue 3 - September 2015*, pages 385–435.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2014. A pdtb-styled end-to-end discourse parser. *Natural Language Engineering*, 20(02):151–184.

Annie Louis, Aravind Joshi, and Ani Nenkova. 2010. Discourse indicators for content selection in summarization. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 147–156. Association for Computational Linguistics.

Thomas Meyer, Najeh Hajlaoui, and Andrei Popescu-Belis. 2015. Disambiguating discourse connectives for statistical machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(7):1184–1197.

George Miller. 1992. Wordnet: a lexical database for english. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):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*, pages 13–16. Association for Computational Linguistics.

Rashmi Prasad and Aravind Joshi. 2008. A discourse-based approach to generating why-questions from texts. In *Proceedings of the Workshop on the Question Generation Shared Task and Evaluation Challenge*, pages 1–3.

Rashmi Prasad, Bonnie Webber, and Aravind Joshi. 2014. Reflections on the penn discourse treebank, comparable corpora, and complementary annotation. *Volume 40, Issue 4 - December 2014*, pages 921–950.

Attapol Rutherford and Nianwen Xue. 2014. Discovering implicit discourse relations through brown cluster pair representation and coreference patterns. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational*

Linguistics, pages 645–654. Association for Computational Linguistics.

Karin Kipper Schuler. 2005. Verbnet: A broad-coverage, comprehensive verb lexicon.

Manfred Stede. 2011. Discourse processing. *Synthesis Lectures on Human Language Technologies*, 4(3):1–165.

Rajen Subba and Barbara Di Eugenio. 2009. An effective discourse parser that uses rich linguistic information. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 566–574. Association for Computational Linguistics.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394. Association for Computational Linguistics.

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

Bonnie Webber, Rashmi Prasad, Alan Lee, and Aravind Joshi. 2016. A discourse-annotated corpus of conjoined vps. *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 22–31.