Learning Connective-based Word Representations for Implicit Discourse Relation Identification

Chloé Braud, Pascal Denis

Learning Connective-based Word Representations
for Implicit Discourse Relation Identification

Chloé Braud
CoAStaL, Dep of Computer Science
University of Copenhagen
University Park 5, 2100 Copenhagen, Denmark
braud@di.ku.dk

Pascal Denis
Magnet Team, INRIA Lille – Nord Europe
59650 Villeneuve dAscq, France
pascal.denis@inria.fr

Abstract

We introduce a simple semi-supervised approach to improve implicit discourse relation identification. This approach harnesses large amounts of automatically extracted discourse connectives along with their arguments to construct new distributional word representations. Specifically, we represent words in the space of discourse connectives as a way to directly encode their rhetorical function. Experiments on the Penn Discourse Treebank demonstrate the effectiveness of these task-tailored representations in predicting implicit discourse relations. Our results indeed show that, despite their simplicity, these connective-based representations outperform various off-the-shelf word embeddings, and achieve state-of-the-art performance on this problem.

1 Introduction

A natural distinction is often made between explicit and implicit discourse relations depending on whether they are lexicalized by a connective or not, respectively. To illustrate, the *Contrast* relation in example (1a) is triggered by the connective *but*, while it is not overtly marked in example (1b).[1] Given the lack of strong explicit cues, the identification of implicit relations is a much more challenging and still open problem. The typically low performance scores for this task also hinder the development of text-level discourse parsers (Lin et al., 2010; Xue et al., 2015): implicit discourse relations

account for around half of the data for different genres and languages (Prasad et al., 2008; Sporleder and Lascarides, 2008; Taboada, 2006; Subba and Di Eugenio, 2009; Soria and Ferrari, 1998; Versley and Gastel, 2013).

(1) a. The house has voted to raise the ceiling to $3.1 trillion, *but* the Senate isn't expected to act until next week at the earliest.

b. That's not to say that the nutty plot of "A Wild Sheep Chase" is rooted in reality. It's imaginative and often funny.

The difficulty of this task lies in its dependence on a wide variety of linguistic factors, ranging from syntax, lexical semantics and also world knowledge (Asher and Lascarides, 2003). In order to deal with this issue, a common approach is to exploit hand-crafted resources to design features capturing lexical, temporal, modal, or syntactic information (Pitler et al., 2009; Park and Cardie, 2012). By contrast, more recent work show that using simple low-dimensional word-based representations, either cluster-based or distributed (aka word embeddings), yield comparable or better performance (Rutherford and Xue, 2014; Braud and Denis, 2015), while dispensing with feature engineering.

While standard low-dimensional word representations appear to encode relevant linguistic information, they have not been built with the specific rhetorical task in mind. A natural question is therefore whether one could improve implicit discourse relation identification by using word representations that are more directly related to the task. The

---

[1]These examples are taken from documents wsj_0008 and wsj_0037, respectively, of the PDTB.

problem of learning good representation for discourse has been recently tackled by Ji and Eisenstein (2014) on the problem of text-level discourse parsing. Their approach uses two recursive neural networks to jointly learn the task and a transformation of the discourse segments to be attached. While this type of joint learning yields encouraging results, it is also computationally intensive, requiring long training times, and could be limited by the relatively small amount of manually annotated data available.

In this paper, we explore the possibility of learning a distributional word representation adapted to the task by selecting relevant rhetorical contexts, in this case discourse connectives, extracted from large amounts of automatically detected connectives along with their arguments. Informally, the assumption is that the estimated word-connective co-occurrence statistics will in effect give us an important insight to the rhetorical function of different words. The learning phase in this case is extremely simple, as it amounts to merely estimating co-occurrence frequencies, potentially combined with a reweighting scheme, between each word appearing in a discourse segment and its co-occurring connective. To assess the usefulness of these connective-based representations,[2] we compare them with pre-trained word representations, like Brown clusters and other word embeddings, on the task of implicit discourse relation identification. Our experiments on the Penn Discourse Treebank (PDTB) (Prasad et al., 2008) show that these new representations deliver improvements over systems using these generic representations and yield state-of-the-art results, and this without the use of other hand-crafted features, thus also alleviating the need for external linguistic resources (like lexical databases). Thus, our approach could be easily extended to resource-poor languages as long as connectives can be reliably identified on raw texts.

Section 2 summarizes related work. In Section 3, we detail our connective-based distributional word representation approach. Section 4 presents the automatic annotation of the explicit examples used to build the word representation. In Section 5, we describe our comparative experiments on the PDTB.

---

[2]Available at `https://bitbucket.org/chloebt/discourse-data`.

2 Related Work

Implicit discourse relation identification has attracted growing attention since the release of the PDTB, the first discourse corpus to make the distinction between explicit and implicit examples. Within the large body of research on this problem, we identify two main strands directly relevant to our work.

2.1 Finding the Right Input Representation

The first work on this task (Marcu and Echihabi, 2002), which pre-dates the release of the PDTB, proposed a simple word-based representation: they use the Cartesian product of words appearing in the two segments. Given the knowledge-rich nature of the task, following studies attempted to exploit various hand-crafted resources and pre-processing systems to enrich their model with information on modality, polarity, tense, lexical semantics, and syntax, possibly combined with feature selection methods (Pitler et al., 2009; Lin et al., 2009; Park and Cardie, 2012; Biran and McKeown, 2013; Li and Nenkova, 2014). Interestingly, Park and Cardie (2012) concluded on the worthlessness of word-based features, as long as hand-crafted linguistic features were used. More recent studies however reversed this conclusion (Rutherford and Xue, 2014; Braud and Denis, 2015), demonstrating that word-based features can be effective provided they were not encoded using the sparse one-hot representation, but instead with a denser one (cluster based or distributed). This paper takes one step further by testing whether learning a simple task-specific, distributional word representation could lead to further improvements.

As noted, some previous work have also attempted to learn discourse-specific representation for the related problem of discourse parsing. Thus, Ji and Eisenstein (2014) reports improvements on the RST Discourse Treebank (Carlson et al., 2001), by jointly learning a combination of the discourse units, represented by bag-of-words in a one-hot encoding, along with the sequence of actions of their shift-reduce parser. Our approach is attractively simpler, since training reduces to collecting frequency counts, and it can easily generate representations for unseen words without having to retrain the whole system.

2.2 Leveraging Explicit Discourse Data

Another line of work, also initiated in (Marcu and Echihabi, 2002), propose to deal with the sparseness of the word pair representation by using additional data automatically annotated using discourse connectives. An appeal of this strategy is that one can easily identify explicit relations in raw data, as performance are high on this task (Pitler et al., 2009) and it is even possible to rely on simple heuristics (Marcu and Echihabi, 2002; Sporleder and Lascarides, 2005; Lan et al., 2013). It has been shown, however, that using explicit examples as additional data for training an implicit relation classifier degrades performance, due to important distribution differences (Sporleder and Lascarides, 2008).

Recent attempts to overcome this issue involve domain adaptation strategies (Braud and Denis, 2014; Ji et al., 2015), sample selection (Rutherford and Xue, 2015; Wang et al., 2012), or multi-task algorithms (Lan et al., 2013). However, it generally involves longer training time since models are built on a massive amount of data, the strategy requiring a large corpus of explicit examples to overcome the noise induced by the automatic annotation strategy. In this paper, we circumvent this problem by using explicit data only for learning our word representations and not for estimating the parameters of our implicit classification model. Some aspects of the present work are similar to Biran and McKeown (2013) in that they also exploit explicit data to compute co-occurrence statistics between word pairs and connectives. But the perspective is reversed, as they represent connectives in the contexts of co-occurring word pairs, with the aim of deriving similarity features between each implicit example and each connective. Furthermore, their approach did not outperform state-of-the-art systems.

3 The Connective Vector Space Model

Our discourse-based word representation model is a simple variant of the standard vector space model (Turney and Pantel, 2010): that is, it represents individual words in specific co-occurring contexts (in this case, discourse connectives) that define the dimensions of the underlying vector space. Our specific choice of contexts was guided by two main considerations. On the one hand, we aim at learning word representations that live in a relatively low-dimensional space, so as to make learning a classification function over that space feasible. The number of parameters of that function grows proportionally with that of the input size. Although there is often a lack of consensus among linguists as to the exact definition of discourse connectives, they nevertheless form a closed class. For English, the PDTB recognizes 100 distinct connectives. On the other hand, we want to learn a vectorial representation that captures relevant aspects of the problem, in this case the rhetorical contribution of words. Adapting Harris (1954)'s famous quote, we make the assumption that words occurring in similar *rhetorical* contexts tend to have similar *rhetorical* meanings. Discourse connectives are by definition strong rhetorical cues. As an illustration, Pitler et al. (2009) found that connectives alone unambiguously predict a single relation in 94% of the PDTB level 1 data. By using connectives as contexts, we are thus linking each word to a relation (or a small set of relations), namely those that can be triggered by this connective. Note that for level 2 relations in the PDTB, the connectives are much more ambiguous (86.77% reported in (Lin et al., 2010)), and it could be also the case if we expand the list of forms considered as connectives for English, or if we try to deal with other languages and domains. We however believe that the set of relations that can be triggered by a connective is limited (not all relations can be expressed by the same connective), and that one attractive feature of our strategy is precisely to keep this ambiguity.

Before turning to the details of how we construct our distributional connective-based model, note that we decided to learn a unique representation for any individual word, irrespective of its position (with)in a particular segment. That is, we represent both arguments of a connective as a single bag of words. Other designs are of course possible: we could directly learn distinct word representation for left and right segment words, or even the pair of words (Conrath et al., 2014), to take into account the fact that some relations are oriented (e.g. *Reason* contains the cause in the first argument and *Result* in the second one). An obvious drawback of these more expressive representations is that they would need much more data to compute a robust estimate of the frequency counts.

Word	but			while			before		
	Freq.	TF-IDF	PPMI-IDF	Freq.	TF-IDF	PPMI-IDF	Freq.	TF-IDF	PPMI-IDF
reality	12	0.0	0.0	13	0.0	0.0	10	0.0	0.0
not	142	0.37	0.36	201	0.18	0.06	0	0.0	0.0
week	0	0.0	0.0	110	0.10	0.04	90	0.12	0.12

Table 1: Illustrative example of association measures between connectives and words.

3.1 Building the Distributional Representation

Our discourse-based representations of words are obtained by computing a matrix of co-occurrence between the words and the chosen contexts. The frequency counts are then weighted in order to highlight relevant associations. More formally, we note $\mathcal{V}$ the set of the n words appearing in the arguments, and $\mathcal{C}$ the set of the m connective contexts. We build the matrix $\mathbf{F}$, of size $n \times m$, by computing the frequency of each element of $\mathcal{V}$ with each element of $\mathcal{C}$. We note $f_{i,j}$ the frequency of the word $w_i \in \mathcal{V}$ appearing in one argument of the connective $c_j \in \mathcal{C}$. We use two standard weighting functions on these raw frequencies: the normalized Term Frequency (TF), eq. (1), and the Positive Pointwise Mutual Information (PPMI), eq. (2), which is a version of the PMI where negative values are ignored (with $p_{i,j}$ the joint probability that the word w_i appears with connective c_j, and $p_{i,*}$ and $p_{*,j}$, relative frequency of resp. w_i and c_j). These two measures are then normalized by multiplying the value by the Inverse Document Frequency (IDF) for a word w_i, eq. (3), as in (Biran and McKeown, 2013). In the final matrices, the ith row corresponds to the m-dimensional vector for the ith word of $\mathcal{V}$. The jth column is a vector corresponding to the jth connective.

$$\text{TF}_{i,j} = \frac{f_{i,j}}{\sum_{k=1}^{n} f_{k,j}} \quad (1)$$

$$\text{PPMI}_{i,j} = \max(0, \log\left(\frac{p_{i,j}}{p_{i,*}\,p_{*,j}}\right)) \quad (2)$$

$$\text{IDF}_i = \log\left(\frac{m}{\sum_{k=1}^{m} f_{i,k}}\right) \quad (3)$$

Table 1 illustrates the weighting of the words using the TF and the PPMI normalized with IDF. For instance, the presence of the negation "not" is positively linked to *Contrast* through *but* and *while* whereas it receives a null or a very small weight with the temporal connective *before*. The final vec-

tor for this word, $< 0.37, 0.18, 0.0 >$ with TF-IDF or $< 0.36, 0.06, 0.0 >$ with PPMI-IDF, is intended to guide the implicit model toward a contrastive relation, thus potentially helping in identifying the relation in example (1b). In contrast, the word "week" is more likely to be found in the arguments of temporal relations that can be triggered by *before* but also *while*, an ambiguity kept in our representation whereas approaches based on using explicit examples as new training data generally choose to annotate them using the most frequent sense associated with the connective, often limiting themselves to the less ambiguous ones (Marcu and Echihabi, 2002; Sporleder and Lascarides, 2008; Lan et al., 2013; Braud and Denis, 2014; Rutherford and Xue, 2015). Finally, a word occuring with all connectives, not discriminant, such as "reality" is associated with a null weight for all dimensions: it thus has no impact on the model.

Since we have 100 connectives for the PDTB, the representation is already of quite low dimensionality. However, it has been shown (Turney and Pantel, 2010) that using a dimensionality reduction algorithm could help capturing the latent dimensions between the words and their contexts and reducing the noise. We thus also test versions with a reduction Components Analysis (PCA) (Jolliffe, 2002).

3.2 Using the Word-based Representation

So far, our distributional framework associates a word with a d-dimensional vector (where $d \leq m$). We now need to represent a pair of arguments (i.e., the spans of text linked by a relation), modeled here as a pair of bags of words. Following (Braud and Denis, 2015), we first sum all word vectors contained in each segment, thus obtaining a d-dimensional vector for each segment. We then combine the two segment vectors to build a composite vector representing the pair of arguments, by ei-

ther concatenating the two segment vectors (leading to a $2d$-dimensional vector) or by computing the Kronecker product between them (leading to a d^2-dimensional vector). Finally, these segment-pair representations will be normalized using the L_2 norm to avoid segment size effects. These will then be used as the input of a classification model, as described in Section 5. Given these combination schemes, it should be clear that despite the fact that each individual word receives a unique vectorial representation irrespective of its position, the parameters of the classification model associated with a given word are likely to be different depending of whether it appears in the left or right segment.

4 Automatic Annotation of Explicit Examples

In order to collect reliable word-connective co-occurrence frequencies, we need a large corpus where the connectives and their arguments have been identified. We therefore rely on automatic annotation of raw data, instead of using the relatively small amount of explicit examples manually annotated in the PDTB (roughly $18,000$ examples). Specifically, we used the *Bllip* corpus[3] composed of news articles from the *LA Times*, the *Washington Post*, the *New York Times* and *Reuters* and containing 310 millions of words automatically POS-tagged.

Identifying the Connectives and their Arguments We have two tasks to perform: identifying the connectives and extracting their arguments.[4] Rather than relying on manually defined patterns to annotate explicit examples (Marcu and Echihabi, 2002; Sporleder and Lascarides, 2008; Rutherford and Xue, 2015), we use two binary classification models inspired by previous works on the PDTB (Pitler and Nenkova, 2009; Lin et al., 2010): the first one identifies the connectives and the second one localizes the arguments between inter- and intra-sentential, an heuristic being then used to decide on the exact boundaries of the arguments.

Discourse connectives are words (e.g., *but*, *since*)

---

[4]Note that contrary to studies using automatically annotated explicit examples as new training data, we do not need to annotate the relation triggered by the connective.

or grammaticalized multi-word expressions (e.g., *as soon as*, *on the other hand*) that may trigger a discourse relation. However, these forms can also appear without any discourse reading, such as *because* in: *He can't sleep because of the deadline*. We thus need to disambiguate these forms between discourse and non discourse readings, a task that has proven to be quite easy on the PDTB (Pitler and Nenkova, 2009). This is the task performed by our first binary classifier: a pattern-matching is used to identify all potential connectives, and the model predicts if they have discourse reading in context.

We then need to extract the arguments of the identified connectives, that is the two spans of text linked by the connective. This latter task has proven to be extremely hard on the PDTB (Lin et al., 2010; Xue et al., 2015) because of some annotation principles that make the possible types of argument very diverse. As first proposed in (Lin et al., 2010), we thus split this task into two subtasks: identifying the relative positions of the arguments and delimiting their exact boundaries.

For an explicit example in the PDTB, one argument, called *Arg2*, is linked to the connective, and thus considered as easy to extract (Lin et al., 2010). The other argument, called *Arg1*, may be located at different places relative to *Arg2* (Prasad et al., 2008): we call intra-sentential the examples where *Arg1* is a clause within the same sentence as *Arg2* (60.9% of the explicit examples in the PDTB), and inter-sentential the other examples, that is *Arg1* is found in the previous sentence, in a non-adjacent previous sentence (9%) or in a following sentence (less than 0.1%). In this work, we build a localization model by only considering these two coarse cases – the example is either intra- or inter-sentential. Note that this distinction is similar to what has been done in (Lin et al., 2010): more precisely, these authors distinguish between "same-sentence" and "previous sentence" and ignore the cases where the *Arg1* is in a following sentence. We rather choose to include them as being also inter-sentential. When the position of *Arg1* has been predicted, an heuristic is in charge of finding the exact boundaries of the arguments.

Here, the problem is that in addition to the variety of locations, the annotators were almost free to choose any boundary for an argument in the PDTB:

an argument can cover only a part of a sentence, an entire sentence or several sentences. Statistical approaches intended to solve this task lead for now to low performance even when complex sequential models are used, and they often rely on the syntactic configurations (Lin et al., 2010; Xue et al., 2015). We thus decided to define an heuristic to perform this task, following the simplifying assumptions also used in previous work since (Marcu and Echihabi, 2002). We assume that: (1) *Arg1* is either in the same sentence as *Arg2* or in the previous one, (2) an argument covers at most one sentence and (3) a sentence contains at most two arguments. As it can be deduced from (1), our final model ignores the finer distinctions one can make for the position of inter-sentential examples (i.e. we never extract *Arg1* from a non-adjacent previous sentence or a following one).

According to these assumptions, once a connective is identified, knowing its localization is almost enough to identify the boundaries of its arguments. More precisely, if a connective is predicted as inter-sentential, then our heuristic picks the entire preceding sentence as *Arg1*, *Arg2* being the sentence containing the connective, according to assumptions (1) and (2). If a connective is predicted as intra-sentential, then the sentence containing the connective is split into two segments – according to (3) –, more precisely, the sentence is split around the connective using the punctuation and making it necessary to have a verb in each argument.

Settings We thus built two models using the PDTB: one to identify the discourse markers (connective *vs* not connective), and one to identify the position of the arguments with respect to the connective (inter- *vs* intra-sentential). The PDTB contains $18,459$ explicit examples for 100 connectives. For both models, we use the same split of the data as in (Lin et al., 2014). The test set contains 923 positive instances of connectives and $2,075$ negative instances, and 546 inter-sentential and 377 intra-sentential examples. Both models are built using a logistic regression model optimized on the development set (see Section 5), and the same simple feature set (Lin et al., 2014; Johannsen and Sgaard, 2013) without syntactic information. With C the connective, F the following word and P the previous one,

our features are: C, P+C, C+F, C-POS[5], P-POS, F-POS, P-POS+C-POS and C-POS+F-POS.

Results Our model identifies discourse connective with a micro-accuracy of 92.9% (macro-F_1 91.5%). These scores are slightly lower than the state-of-the-art in micro-accuracy, but high enough to rely on this annotation. When applying our model to the *Bllip* data, we found 4 connectives that correspond to no examples. We thus have examples for only 96 connectives. For distinguishing between inter- and intra-sentential examples, we get a micro-accuracy of 96.1% (macro-F_1 96.0), with an F_1 of 96.7 for the intra- and 95.3 for the inter-sentential class, again close enough to the state-of-the-art (Lin et al., 2014).

Coverage Using these models on *Bllip*, we are able to extract around 3 million connectives, along with their arguments. Our word representation has a large vocabulary (see Table 2) compared to existing off-the-shelf word vectors, with only $2,902$ out of vocabulary (OOV) tokens in set of implicit relations.[6]

	# words	# OOV
HLBL	$246,122$	$5,439$
CnW	$268,810$	$5,638$
Brown	$247,339$	$5,413$
H-PCA	$178,080$	$7,042$
Bllip	$\mathbf{422,199}$	$\mathbf{2,902}$

Table 2: Lexicon coverage for *Brown* clusters (Brown et al., 1992), Collobert and Weston (*CnW*) (Collobert and Weston, 2008) and hierarchical log-bilinear embeddings (*HLBL*) (Mnih and Hinton, 2007) using the implementation in (Turian et al., 2010), Hellinger PCA (*H-PCA*) (Lebret and Collobert, 2014) and our connective-based representation (*Bllip*).

5 Experiments

Our experiments investigate the relevance of our connective-based representations for implicit discourse relation identification, recast here as multi-class classification problem. That is, we aim at evaluating the usefulness of having a word representation linked to the task, compared to using generic

---

[5]The connective POS is either the node covering the connective, or the POS of its first word if no such node exists.

[6]Training and development sets, only.

Relation	Train	Dev	Test
Temporal	665	93	68
Contingency	3,281	628	276
Comparison	1,894	401	146
Expansion	6,792	1,253	556
Total	12,632	2,375	1,046

Table 3: Number of examples in train, dev, test.

word representations (either one-hot, cluster-based or distributed), and whether they encode all the information relevant to the task, thus comparing systems with or without additional hand-crafted features.

5.1 Data

The PDTB (Prasad et al., 2008) is the largest corpus annotated for discourse relations, formed by newspaper articles from the Wall Street Journal. It contains 16,053 pairs of spans of text annotated with one or more implicit relations. The relation set is organized in a three-level hierarchy. We focus on the level 1 coarse-grained relations and keep only the first relation annotated. We use the most spread split of the data, used in (Rutherford and Xue, 2014; Rutherford and Xue, 2015; Braud and Denis, 2015) among others, that is sections 2-20 for training and 21-22 for testing. The other sections are used for development. The number of examples per relation is reported in Table 3. It can be seen that the dataset is highly imbalanced, with the relation *Expansion* accounting for more than 50% of the examples.

5.2 Settings

Feature Set Our main features are based on the words occurring in the arguments. We test simple baselines using raw tokens. The first one uses the Cartesian product of the tokens, a feature template, generally called "Word pairs", used in most of the previous study for this task as in (Marcu and Echihabi, 2002; Pitler et al., 2009; Lin et al., 2011; Braud and Denis, 2015; Ji et al., 2015). It is the sparsest representation one can build from words, and it corresponds to using the combination scheme based on the Kronecker product to combine the one-hot vectors representing each word. We also report results with a less sparse version where the vectors are combined using concatenation.

We also compare our systems to previous approaches that make use of word based representations but not linked to the task. We implement the systems proposed in (Braud and Denis, 2015) in multiclass, that is using the *Brown* clusters (Brown et al., 1992), the Collobert and Weston (Collobert and Weston, 2008) and the hierarchical log-bilinear embeddings (Mnih and Hinton, 2007) using the implementation in (Turian et al., 2010)[7], and the *HPCA* (Lebret and Collobert, 2014)[8]. We use the combination schemes described in Section 3 to build vector representations for pairs of segments. For these systems and ours, using the connective-based representations, the dimensionality of the final model depends on the number of dimensions d of the representation used and on the combination scheme – the concatenation leading to $2d$ dimensions and the Kronecker product to d^2.

All the word representations used – the off-the-shelf representations as well as our connective-based representation (see Section 4) – are solely or mainly trained on newswire data, thus on the same domain as our evaluation data. The *CnW* embeddings we use in this paper, with the implementation in (Turian et al., 2010), as well as the *HLBL* embeddings have been obtained using the RCV1 corpus, that is one year of Reuters English newswire. The *H-PCA* have been built on the Wikipedia, the Reuters corpus and the Wall street Journal. We thus do not expect any out-of-domain issue when using these representations.

Finally, we experiment with additional features proposed in previous studies and well described in (Pitler et al., 2009; Park and Cardie, 2012): production rules[9], information on verbs (average verb phrases length and Levin classes), polarity (Wilson et al., 2005), General Inquirer tags (Stone and Kirsh, 1966), information about the presence of numbers and modals, and first, last and first three words. We concatenate these features to the ones built using word representations.

---

[7]http://metaoptimize.com/projects/wordreprs/
[8]http://lebret.ch/words/
[9]We use the gold standard parses provided in the Penn Treebank (Marcus et al., 1993).

Model We train a multinomial multiclass logistic regression model.[10] In order to deal with the class imbalance issue, we use a sample weighting scheme where each instance has a weight inversely proportional to the frequency of the class it belongs to.

Parameters We optimize the hyper-parameters of the algorithm, that is the regularization norm (L1 or L2), and the strength of the regularization $C \in \{0.001, 0.005, 0.01, 0.1, 0.5, 1, 5, 10, 100\}$. When using additional features or one-hot sparse encodings over the pairs of raw tokens, we also optimize a filter on the features by defining a frequency cutoff $t \in \{1, 2, 5, 10, 15, 20\}$. We evaluate the unsupervised representations with different number of dimensions. We test versions of the *Brown* clusters with 100, 320, 1,000 and 3,200 clusters, of the Collobert and Weston embeddings with 25, 50, 100 and 200 dimensions, of the hierarchical log-bilinear embeddings with 50 and 100 dimensions, and of the Hellinger PCA with 50, 100 and 200 dimensions. Finally, the distributional representations of words based on the connective are built using either no PCA – thus corresponding to 96 dimensions–, or a PCA[11] keeping the first k dimensions with $k \in \{2, 5, 10, 50\}$.[12] We optimize both the hyperparameters of the algorithm and the number of dimensions of the unsupervised representation on the development set based on the macro-F_1 score, the most relevant measure to track when dealing with imbalanced data.

5.3 Results

Our results are summarized in Table 4. Using our connective-based word representation allows improvements of above 2% in macro-F_1 over the baseline systems based on raw tokens (*One-hot*), the competitive systems using pre-trained representations (*Brown* and *Embed.*) and the state-of-the-art results in terms of macro-F_1 (R&X 15). These improvements demonstrate the efficiency of the representation for this task.

We found that using an unsupervised word representation generally leads to improvements over the

Representation	Macro-F_1	Acc.
One-hot $\otimes$	39.0	48.6
One-hot $\oplus$	40.2	50.2
Best *Brown* $\otimes$	37.5	50.6
Best *Brown* $\oplus$	40.6	51.2
Best *Embed.* $\otimes$	41.0	51.7
Best *Embed.* $\oplus$	41.6	50.1
Best dense + add feat.	40.8	51.2
Bllip TF-IDF $\otimes$	41.4	51.0
Bllip TF-IDF $\oplus$	40.1	50.0
Bllip PPMI-IDF $\otimes$	38.9	48.2
Bllip PPMI-IDF $\oplus$	**42.2**[*]	52.5
Best *Bllip* + add feat.	**42.8**[*]	51.7
R&X 15	40.5	57.1

Table 4: Results for multiclass experiments. R&X 15 are the scores reported in (Rutherford and Xue, 2015) ; *One-hot*: one-hot encoding of raw tokens ; *Brown* and *Embed.*: pre-trained representations ; *Bllip*: connective based representation. [*] $p \leq 0.1$ compared to *One-hot* $\otimes$ with t-test and Wilcoxon.

use of raw tokens (*One-hot*), a conclusion in line with the results reported in (Braud and Denis, 2015) for binary systems. However, contrary to their findings, in multiclass, the best results are not obtained using the *Brown* clusters, but rather the dense, real valued representations (*Embed.* and *Bllip*). Furthermore, concerning the combination schemes, the concatenation ($\oplus$) generally outperforms the Kronecker product ($\otimes$), in effect favoring lower dimensional models.

More importantly, the distributional representations based on connectives (*Bllip*) allows performance at least similar or even better than those obtained with the other dense representations unconnected to the task (*Embed.*). While simply based on weighted co-occurrence counts, thus really easy and fast to build, these representations generally outperform the ones learned using neural networks (see CnW and HLBL in Figure 1). Besides, our second best representation is also distributional, namely HPCA (see Figure 1). These result are thus in line with the conclusions in (Lebret and Collobert, 2014) for other NLP tasks: distributional representations, while simpler to obtain, may allow similar results than distributed ones.

---

[10]http://scikit-learn.org/dev/index.html.

[11]Implemented in scikit-learn, applied with default settings.

[12]Keeping resp. 11.3%, 36.6%, 56.2% or 95.3% of the variance of the data.
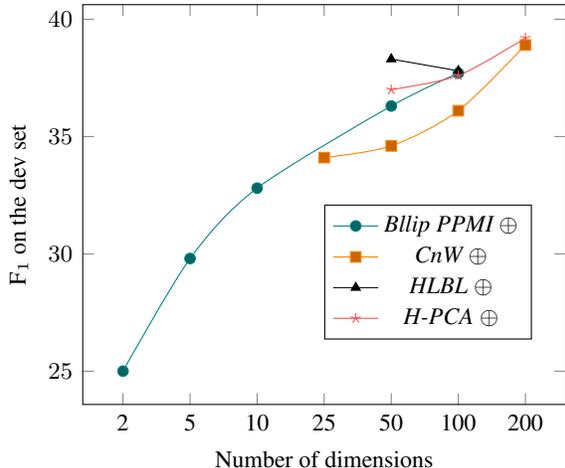
Figure 1: F_1 scores on dev against the number of dimensions.

Our best results with *Bllip* are obtained without the use of a dimensionality reduction method, thus keeping the 96 dimensions corresponding to the connectives identified in the raw data. Our new word representation like the other low-dimensional ones yield higher scores as one increases the number of dimensions (see Figure 1). This could be a limitation of our strategy, since the number of connectives in the PDTB is fixed. However, one could easily expand our model to include additional lexical elements that might have a rhetorical function such as modals or specific expressions such as *one reason is*.

We also tested the addition of hand-crafted features traditionally used for the task. We found that, either using a pre-trained word representation or our representation based on connectives, adding these features leads to small or even no improvements and suggest that these representations already encode the information provided by these features. This conclusion has however to be nuanced: when looking at the scores per relation reported in Table 5, the use of the connective based word representation alone allows the best performance for *Temporal* and *Contingency*, but the addition of new features dramatically increase the scores for *Comparison* showing that some information are missing for this relation. Moreover, this relation is the one taking the most advantage of the addition of explicit data in (Rutherford and Xue, 2015), demonstrating that these data could probably provide even more information than the ones we leverage through our representations.

Finally, our results are similar or even better than those reported in (Rutherford and Xue, 2015) in terms of macro-F_1. Our systems correspond however to a lower micro-accuracy. Looking at the scores per relation in Table 5, we found that we obtain better results for all the relations except *Expansion*, the most represented, which could explain the loss in accuracy. It is noteworthy that we generally obtain better results even without the additional features used in this work. Moreover, our systems requires lower training time (since we only train on implicit examples) and alleviate the need for the sample selection strategy used to deal with the distribution differences between the two types of data.

| | *Bllip* PPMI-IDF $\oplus$ | | *Bllip + add feat* | | R&X 15 | |
Rel	Prec	F_1	Prec	F_1	Prec	F_1
Temp	23.0	**29.9**	23.7	27.9	38.5	14.7
Cont	49.6	**47.1**	46.7	46.3	49.3	43.9
Comp	35.9	27.7	35.0	**34.3**	44.9	34.2
Exp	62.8	64.0	63.7	62.6	61.4	**69.1**

Table 5: Scores per relation for multiclass experiments, "R&X 15" are the scores reported in (Rutherford and Xue, 2015).

6 Conclusion

We presented a new approach to leverage information from explicit examples for implicit relation identification. We showed that building distributional representations linked to the task through connectives allows state-of-the-art performance and alleviates the need for additional features. Future work includes extending the representations to new contexts – such as the Alternative Lexicalization annotated in the PDTB, the modals or some adverbs – using more sophisticated weighting schemes (Lebret and Collobert, 2014) and testing this strategy for other languages and domains.

Acknowledgements

References

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Cambridge University Press.

Or Biran and Kathleen McKeown. 2013. Aggregated word pair features for implicit discourse relation disambiguation. In *Proceedings of ACL*.

Chloé Braud and Pascal Denis. 2014. Combining natural and artificial examples to improve implicit discourse relation identification. In *Proceedings of COLING*.

Chloé Braud and Pascal Denis. 2015. Comparing word representations for implicit discourse relation classification. In *Proceedings of EMNLP*.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18:467–479.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2001. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of ICML*.

Juliette Conrath, Stergos Afantenos, Nicholas Asher, and Philippe Muller. 2014. Unsupervised extraction of semantic relations using discourse cues. In *Proceedings of Coling*.

Zellig S. Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Yangfeng Ji and Jacob Eisenstein. 2014. Representation learning for text-level discourse parsing. In *Proceedings of ACL*.

Yangfeng Ji, Gongbo Zhang, and Jacob Eisenstein. 2015. Closing the gap: Domain adaptation from explicit to implicit discourse relations. In *Proceedings of EMNLP*.

Anders Johannsen and Anders Sgaard. 2013. Disambiguating explicit discourse connectives without oracles. In *Proceedings of IJCNLP*.

Ian Jolliffe. 2002. *Principal component analysis*. Wiley Online Library.

Man Lan, Yu Xu, and Zhengyu Niu. 2013. Leveraging synthetic discourse data via multi-task learning for implicit discourse relation recognition. In *Proceedings of ACL*.

Rémi Lebret and Ronan Collobert. 2014. Word emddeddings through Hellinger PCA. In *Proceedings of ACL*.

Junyi Jessy Li and Ani Nenkova. 2014. Reducing sparsity improves the recognition of implicit discourse relations. In *Proceedings of SIGDIAL*.

Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. 2009. Recognizing implicit discourse relations in the Penn Discourse Treebank. In *Proceedings of EMNLP*.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2010. A PDTB-styled end-to-end discourse parser. Technical report, National University of Singapore.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2011. Automatically evaluating text coherence using discourse relations. In *Proceedings of ACL-HLT*.

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

Daniel Marcu and Abdessamad Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *Proceedings of ACL*.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Andriy Mnih and Geoffrey Hinton. 2007. Three new graphical models for statistical language modelling. In *Proceedings of ICML*.

Joonsuk Park and Claire Cardie. 2012. Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of SIGDIAL Conference*.

Emily Pitler and Ani Nenkova. 2009. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP*.

Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. In *Proceedings of ACL-IJCNLP*.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse Treebank 2.0. In *Proceedings of LREC*.

Attapol Rutherford and Nianwen Xue. 2014. Discovering implicit discourse relations through Brown cluster pair representation and coreference patterns. In *Proceedings of EACL*.

Attapol Rutherford and Nianwen Xue. 2015. Improving the inference of implicit discourse relations via classifying explicit discourse connectives. In *Proceedings of NAACL-HLT*.

Claudia Soria and Giacomo Ferrari. 1998. Lexical marking of discourse relations - some experimental findings. In *Proceedings of the ACL Workshop on Discourse Relations and Discourse Markers*.

Caroline Sporleder and Alex Lascarides. 2005. Exploiting linguistic cues to classify rhetorical relations. In *Proceedings of RANLP-05*.

Caroline Sporleder and Alex Lascarides. 2008. Using automatically labelled examples to classify rhetorical relations: An assessment. *Natural Language Engineering*, 14:369–416.

Philip J. Stone and John Kirsh. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press.

Rajen Subba and Barbara Di Eugenio. 2009. An effective discourse parser that uses rich linguistic information. In *Proceedings of ACL-HLT*.

Maite Taboada. 2006. Discourse markers as signals (or not) of rhetorical relations. *Journal of Pragmatics*, 38:567–592.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of ACL*.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning : Vector space models of semantics. *Journal of Artificial Intelligence Research*, pages 141–188.

Yannick Versley and Anna Gastel. 2013. Linguistic tests for discourse relations in the TüBa-D/Z corpus of written German. *Dialogue & Discourse*, 4(2):142–173.

Xun Wang, Sujian Li, Jiwei Li, and Wenjie Li. 2012. Implicit discourse relation recognition by selecting typical training examples. In *Proceedings of COLING 2012: Technical Papers*.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT-EMNLP*.

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 CoNLL*.