Aligning Predicates across Monolingual Comparable Texts using Graph-based Clustering

Michael Roth and **Anette Frank**
Department of Computational Linguistics
Heidelberg University
Germany
{mroth,frank}@cl.uni-heidelberg.de

Abstract

Generating coherent discourse is an important aspect in natural language generation. Our aim is to learn factors that constitute coherent discourse from data, with a focus on how to realize predicate-argument structures in a model that exceeds the sentence level. We present an important subtask for this overall goal, in which we align predicates across comparable texts, admitting partial argument structure correspondence. The contribution of this work is two-fold: We first construct a large corpus resource of comparable texts, including an evaluation set with manual predicate alignments. Secondly, we present a novel approach for aligning predicates across comparable texts using graph-based clustering with Mincuts. Our method significantly outperforms other alignment techniques when applied to this novel alignment task, by a margin of at least 6.5 percentage points in F_1-score.

1 Introduction

Discourse coherence is an important aspect in natural language generation (NLG) applications. A number of theories have investigated coherence inducing factors. A prominent example is *Centering Theory* (Grosz et al., 1995), which models local coherence by relating the choice of referring expressions to the importance of an entity at a certain stage of a discourse. A data-driven model based on this theory is the *entity-based approach* by Barzilay and Lapata (2008), which models coherence phenomena by observing sentence-to-sentence transitions of entity occurrences.

Barzilay and Lapata show that their approach can discriminate between a coherent and a non-coherent set of ordered sentences. However, their model is not able to generate alternative entity realizations by itself. Furthermore, the entity-based approach only investigates realization patterns for individual entities in discourse in terms of core grammatical functions. It does not investigate the interplay between entity transitions and realization patterns for full-fledged semantic structures. This interplay, however, is an important factor for a semantics-based, generative model of discourse coherence.

The main hypothesis of our work is that we can automatically learn context-specific realization patterns for predicate argument structures (PAS) from a semantically parsed corpus of comparable text pairs. Our assumption builds on the success of previous research, where comparable and parallel texts have been exploited for a range of related learning tasks, e.g., unsupervised discourse segmentation (Barzilay and Lee, 2004) and bootstrapping semantic analyzers (Titov and Kozhevnikov, 2010).

For our purposes, we are interested in finding corresponding PAS across comparable texts that are known to talk about the same events, and hence involve the same set of underlying event participants. By aligning predicates in such texts, we can investigate the factors that determine discourse coherence in the realization patterns for the involved arguments. These include the specific forms of argument realization, as a pronoun or a specific type of referential expression, as studied in prior work in NLG (Belz et al., 2009, inter alia). The specific set-up we examine, however, allows us to further investi-

Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 171–182, Jeju Island, Korea, 12–14 July 2012. ©2012 Association for Computational Linguistics

gate the factors that govern the *non-realization* of an argument position, as a special form of coherence inducing element in discourse. Example (1), extracted from our corpus of aligned texts,illustrates this point: Both texts report on the same event of locating victims in an avalanche. While (1.a) explicitly talks about the location of this event, the role remains implicit in the second sentence of (1.b), given that it can be recovered from the preceding sentence. In fact, realization of this argument role would impede the fluency of discourse by being overly repetitive.

(1) a. ...The official said that [no bodies]$_{\text{Arg1}}$ had been recovered [from the avalanches]$_{\text{Arg2}}$ which occurred late Friday in the Central Asian country near the Afghan border some 300 kilometers (185 miles) southeast of the capital Dushanbe.

b. Three other victims were trapped *in an avalanche* in the village of Khichikh. [None of the victims bodies]$_{\text{Arg1}}$ have been found [ ]$_{\text{Argm-loc}}$.

This phenomenon clearly relates to the problem of discourse-linking of implicit roles, a very challenging task in discourse processing.[1] In our work, we consider this problem from a content-based generation perspective, concentrating on the discourse factors that allow for the omission of a role.

Thus, our aim is to identify comparable predications across aligned texts, and to study the discourse coherence factors that determine the realization patterns of arguments in the respective discourses. This can be achieved by considering the full set of arguments that can be recovered from the aligned predications. This paper focuses on the first of these tasks, henceforth called *predicate alignment*.[2]

In line with data-driven approaches in NLP, we automatically align predicates in a suitable corpus of paired texts. The induced alignments will (i) serve to identify events described in both comparable texts, and (ii) provide information about the underlying argument structures and how they are realized in each context to establish a coherent discourse. We investigate a graph-based clustering method for inducing such alignments as clustering provides a suitable framework to implicitly relate alignment decisions to one another, by exploiting global information encoded in a graph.

The remainder of this paper is structured as follows: In Section 2, we discuss previous work in related tasks. Section 3 describes our task and a suitable data set. Section 4 introduces a graph-based clustering model using Mincuts for the alignment of predicates. Section 5 outlines the experiments and presents evaluation results. Finally, we conclude in Section 6 and discuss future work.

2 Related Work

The task of aligning words in general has been studied extensively in previous work, for example as part of research in statistical machine translation (SMT). Typically, alignment models in SMT are trained by observing and (re-)estimating co-occurrence counts of word pairs in parallel sentences (Brown et al., 1993). The same methods have also been applied in monolingual settings, for example to align words in paraphrases (Cohn et al., 2008). In contrast to traditional word alignment tasks, our focus is not on pairs of isolated sentences but on aligning predicates within the discourse contexts in which they are situated. Furthermore, text pairs for our task should not be strictly parallel as we are specifically interested in the impact of different discourse contexts. In Section 5, we will show that this particular setting indeed constitutes a more challenging task compared to traditional word alignment in parallel or paraphrasing sentences.

Another set of related tasks is found in the area of textual inference. Since 2006, there have been regular challenges on the task of Recognizing Textual Entailment (RTE). In the original task description, Dagan et al. (2006) define *textual entailment* "as a directional relationship between pairs of text expressions, denoted by T - the entailing 'Text' -, and H - the entailed 'Hypothesis'. (...) T *entails* H if the meaning of H can be inferred from the meaning of T, as would typically be interpreted by people." Although this relation does not necessarily require the presence of corresponding predicates, previous work by MacCartney et al. (2008) shows that word alignments can serve as a good indicator of entailment.

[1] See the recent SemEval 2010 task: *Linking Events and their Participants in Discourse*, (Ruppenhofer et al., 2010).

[2] Note that we provide details regarding the construction of a suitable data set and further examples involving non-realized arguments in a complementary paper (Roth and Frank, 2012).

As a matter of fact, the same holds true for the task of detecting paraphrases. In contrast to RTE, this latter task requires bi-directional entailments, i.e., each of the two phrases must entail the other. Wan et al. (2006) show that a simple approach solely based on word (and lemmatized n-gram) overlap can already achieve an F_1-score of up to 83% for detecting paraphrases in the Microsoft Research Paraphrase Corpus (Dolan and Brockett, 2005, MSRPC). In fact, this is just 0.6% points below the state-of-the-art results recently reported by Socher et al. (2011).

The MSRPC and data sets from the first RTE challenges only consisted of isolated pairs of sentences. The Fifth PASCAL Recognizing Textual Entailment Challenge (Bentivogli et al., 2009) introduced a "Search Task", where entailing sentences for a hypothesis have to be found in a set of full documents. This new task first opened the doors for assessing the role of discourse (Mirkin et al., 2010a; Mirkin et al., 2010b) in RTE. However, this setting is still limited as discourse contexts are only provided for the entailing part (T) of each text pair but not for the hypothesis H.

A further task related to ours is the detection of event coreference. The goal of this task is to identify all mentions of the same event within a document and, in some settings, also across documents. However, the task setting is typically more restricted than ours in that its focus lies on identical events/references (cf. Walker et al. (2006), Weischedel et al. (2011), inter alia). In particular, verbalizations of different aspects of an event (e.g., 'buy'–'sell', 'kill'–'die', 'recover'–'find') are generally not linked in this paradigm. In contrast to coreference methods that identify chains of events, we are interested in pairs of corresponding predicates (and their argument structure), for which we can observe alternative realizations in discourse.

3 Aligning Predicates Across Texts

This section summarizes how we built a large corpus of comparable texts, as a basis for the *predicate alignment* task. We motivate the choice of the corpus and present a strategy for extracting comparable text pairs. Subsequently, we report on the preparation of an evaluation data set with manual predicate alignments across the paired texts. We conclude this section with an example that showcases the potential of using aligned predicates for the study of coherence phenomena. More detailed information regarding corpus creation, annotation guidelines and additional examples illustrating the potential of this corpus can be found in Roth and Frank (2012).

3.1 Corpus Creation

The goal of our work is to investigate coherence factors for argument structure realization, using comparable texts that describe the same events, but that include variation in textual presentation. This requirement fits well with the news domain, for which we can trace varying textual sources that describe the same underlying events. The English Gigaword Fifth Edition (Parker et al., 2011) corpus (henceforth just *Gigaword*) is one of the largest corpus collections for English. It comprises a total of 9.8 million newswire articles from seven distinct sources.

In previous work (Roth and Frank, 2012), we introduced *GigaPairs*, a sub-corpus extracted from Gigaword that includes over 160,000 pairs of newswire articles from distinct sources. GigaPairs has been derived from Gigaword using the pairwise similarity method on headlines presented by Wubben et al. (2009). In addition to calculating the similarity of news titles, we impose an additional date constraint to further increase the precision of extracted pairs of texts. Random inspection of about 100 documents revealed only two texts describing different events. Overall, we extracted 167,728 document pairs containing a total of 50 million word tokens. Each document in this corpus consists of up to 7.564 words with a mean and median of 301 and 213 words, respectively. All texts have been pre-processed using MATE tools (Björkelund et al., 2010; Bohnet, 2010), a pipeline of NLP modules including a state-of-the-art semantic role labeler that computes PropBank/NomBank annotations (Palmer et al., 2005; Meyers et al., 2008).

3.2 Gold Standard Annotation

We selected 70 text pairs from the GigaPairs corpus for manual predicate alignment. All document pairs were randomly chosen with the constraint that each text consists of 100 to 300 words.[3] Predi-

[3] This constraint is satisfied by 75.3% of all documents in GigaPairs.

cates identified by the semantic parser are provided as pre-labeled annotations for alignment. We asked two students[4] to tag corresponding predicates across each text pair. Following standard practice in word alignment tasks (cf. Cohn et al. (2008)) the annotators were instructed to distinguish between *sure* and *possible* alignments, depending on how certainly, in their opinion, two predicates describe verbalizations of the same event. The following examples show predicate pairings marked as sure (2) and as possible alignments (3).

(2) a. The regulator ruled on September 27 that Nasdaq too was qualified to bid for OMX [. . .]
    b. The authority [. . .] had already approved a similar application by Nasdaq.

(3) a. Myanmar's military government said earlier this year it has released some 220 political prisoners [. . .]
    b. The government has been regularly releasing members of Suu Kyi's National League for Democracy party [. . .]

In total, the annotators (A/B) aligned 487/451 sure and 221/180 possible alignments with a Kappa score (Cohen, 1960) of 0.86.[5] For the construction of a gold standard, we merged the alignments from both annotators by taking the union of all possible alignments and the intersection of all sure alignments. Cases which involved a sure alignment on which the annotators disagreed were resolved in a group discussion with the first author.

We split the final corpus into a development set of 10 document pairs and a test set of 60 document pairs. The test set contains a total of 3,453 predicates (1,531 nouns and 1,922 verbs). Its gold standard annotation consists of 446 sure and 361 possible alignments, which corresponds to an average of 7.4 sure (6.0 possible) alignments per document pair. Most of the gold alignments (82.4%) are between predicates of the same part-of-speech (242 noun and 423 verb pairs). A total of 383 gold alignments (47.5%) have been annotated between predicates with identical lemma form. Diverging numbers of realized arguments can be observed in 320 pairs (39.7%).

3.3 Potential for Discourse Coherence

This section presents an example of an aligned predicate pair from our development set that illustrates the potential of aggregating corresponding PAS across comparable texts. The example represents one of eleven cases involving unrealized arguments that can be found in our development set of only ten document pairs.

(4) a. The Chadians said they$_{\text{Arg0}}$ had fled in fear of their lives.
    b. The United Nations says some 20,000 refugees$_{\text{Arg0}}$ have fled into Cameroon$_{\text{Arg1}}$.

In both sentences, the Arg0 role of the predicate flee is filled, but Arg1 (here: the goal) has not been realized in (4.a). However, sentence (4.a) is still part of a coherent discourse, as a role filler for the omitted argument can be inferred from the preceding context. For the goal of our work, we are interested in factors that license such omissions of an argument. Potential factors on the discourse level include the information status of the entity filling an argument position, and its salience at the corresponding point in discourse. Roth and Frank (2012) discuss additional examples that demonstrate the importance of factors on further linguistic levels, e.g., lexical choice of predicates and their syntactic realization.

In the example above, the aggregation of aligned PAS presents an effective means to identify appropriate fillers for unrealized roles. Hence, we can utilize each such pair as one positive and one negative training instance for a model of discourse coherence that controls the omissibility of arguments. In what follows, we introduce an alignment approach that can be used to automatically acquire more training data using the entire GigaPairs corpus.

4 Model

For the automatic induction of predicate alignments across texts, we opt for an unsupervised graph-based clustering method. In this section, we first define a graph representation for pairs of documents. In particular, predicates are represented as nodes in such a graph and similarities between predicates as edges. We then proceed to describe various similarity measures that can be used to identify similar predicate instances. Finally, we introduce the clustering algorithm that we apply to graphs (representing pairs of

[4] Both annotators are students in computational linguistics, one undergraduate (A) and one postgraduate (B) student.

[5] Following Brockett (2007), we computed agreement on labeled annotations, including unaligned predicate pairs as an additional *null* category.

documents) in order to induce alignments between corresponding predicates.

4.1 Graph representation

We build a bipartite graph representation for each pair of texts, using as vertices the predicate argument structures assigned in pre-processing (cf. Section 3.1). We represent each predicate as a node and integrate information about arguments only implicitly. Given the sets of predicates P_1 and P_2 of two comparable texts T_1 and T_2, respectively, we formally define an undirected graph G_{P_1,P_2} as follows:

$$G_{P_1,P_2} = \langle V, E \rangle \quad \text{where} \quad \begin{array}{l} V = P_1 \cup P_2 \\ E = P_1 \times P_2 \end{array} \qquad (1)$$

Edge weights. We specify the edge weight between two nodes representing predicates $p_1 \in P_1$ and $p_2 \in P_2$ as a weighted linear combination of four similarity measures described in the next section: *WordNet* and *VerbNet* similarity, *Distributional* similarity and *Argument* similarity.

$$\begin{array}{rl} w_{p_1 p_2} = & \lambda_1 * \text{sim}_{\text{WN}}(p_1, p_2) \\ + & \lambda_2 * \text{sim}_{\text{VN}}(p_1, p_2) \\ + & \lambda_3 * \text{sim}_{\text{Dist}}(p_1, p_2) \\ + & \lambda_4 * \text{sim}_{\text{Arg}}(p_1, p_2) \end{array} \qquad (2)$$

Initially we set all weighting parameters $\lambda_1 \ldots \lambda_4$ to have uniform weights by default. In Section 5, we define an optimized weighting setting for the individual similarity measures.

4.2 Similarity Measures

We employ a number of similarity measures that make use of complementary information that is type-based ($\text{sim}_{\text{WN/VN/Dist}}$) or token-based ($\text{sim}_{\text{Arg}}$).[6] Given two lemmatized predicates p_1, p_2 and their set of arguments $A_1 = \text{args}(p_1)$, $A_2 = \text{args}(p_2)$, we define the following measures.

WordNet similarity. Given all pairs of synsets s_1, s_2 that contain the predicates p_1, p_2, respectively, we compute the maximal similarity using the information theoretic measure described in Lin (1998). Our implementation exploits the WordNet hierarchy (Fellbaum, 1998) to find the synset of the least common subsumer (lcs) and uses the pre-computed Information Content (IC) files from Pedersen et al. (2004) to compute Lin's measure:

$$\text{sim}_{\text{WN}}(p_1, p_2) = \frac{IC(\text{lcs}(s_1, s_2))}{IC(s_1) * IC(s_2)} \qquad (3)$$

In order to compute similarities between verbal and nominal predicates, we further use derivation information from NomBank (Meyers et al., 2008): if a noun represents a nominalization of a verbal predicate, we resort to the corresponding verb synset. If no relation can be found between two predicates, we set a default value of $sim_{\text{WN}} = 0$. This applies in particular to all cases that involve a predicate not present in WordNet.

VerbNet similarity. To overcome systematic problems with the WordNet verb hierarchy (cf. Richens (2008)), we further compute similarity between verbal predicates using VerbNet (Kipper et al., 2008). Verbs in VerbNet are categorized into semantic classes according to their syntactic behavior. A class C can recursively embed sub-classes $C_s \in sub(C)$ that represent finer semantic and syntactic distinctions. We define a simple similarity function that defines fixed similarity scores between 0 and 1 for pairs of predicates p_1, p_2 depending on their relatedness within the VerbNet class hierarchy:

$$\text{sim}_{\text{VN}}(p_1, p_2) = \begin{cases} 1.0 & \text{if } \exists C : p_1, p_2 \in C \\ 0.8 & \text{if } \exists C, C_s : C_s \in sub(C) \\ & \quad \wedge \; p_1, p_2 \in C \cup C_s \\ 0.0 & \text{else} \end{cases} \qquad (4)$$

Distributional similarity. As some predicates may not be covered by the WordNet and VerbNet hierarchies, we additionally calculate similarity based on distributional meaning in a semantic space (Landauer and Dumais, 1997). Following the traditional bag-of-words approach that has been applied in related tasks (Guo and Diab, 2011; Mitchell and Lapata, 2010), we consider the 2,000 most frequent context words $c_1, \ldots, c_{2000} \in C$ as dimensions of a vector space and define predicates as vectors using their Pointwise Mutual Information (PMI):

$$\vec{p} = (\text{PMI}(p, c_1), \ldots, \text{PMI}(p, c_{2000}) \qquad (5)$$

[6] All token-based frequency counts (i.e., $freq()$ and $idf()$) are computed over all documents from the AFP and APW parts of the English Gigaword Fifth Edition.

$$\text{with} \quad \text{PMI}(x, y) = \frac{\text{freq}(x, y)}{\text{freq}(x) * \text{freq}(y)}$$

Given the vector representations of two predicates, we calculate their similarity as the cosine of the angle between the two vectors:

$$\text{sim}_{\text{Dist}}(p_1, p_2) = \frac{\vec{p_1} \cdot \vec{p_2}}{|\vec{p_1}| * |\vec{p_2}|} \tag{6}$$

Argument similarity. While the previous similarity measures are purely type-based, *argument similarity* integrates token-based, i.e., discourse-specific, similarity information about predications by taking into account the similarity of their arguments. This measure calculates the association between the arguments A_1 of the first and the arguments A_2 of the second predicate by determining the ratio of overlapping words in both argument sets.

$$\text{sim}_{\text{Arg}}(p_1, p_2) = \frac{\sum_{w \in A_1 \cap A_2} \text{idf}(w)}{\sum_{w \in A_1} \text{idf}(w) + \sum_{w \in A_2} \text{idf}(w)} \tag{7}$$

In order to give higher weight to (rare) content words, we weight each word by its Inverse Document Frequency (IDF), which we calculate over all documents d from the AFP and APW sections of the Gigaword corpus:

$$\text{idf}(w) = \log \frac{|D|}{|\{d : w \in D|\}} \tag{8}$$

Normalization. In order to make the outputs of all similarity measures comparable, we normalize their value ranges on the development set to have a mean and standard deviation of 1.0.

4.3 Mincut-based Clustering

Our graph clustering method uses minimum cuts (or *Mincut*) in order to partition the bipartite text graph into clusters of aligned predicates. A Mincut operation divides a given graph into two disjoint subgraphs. Each minimum cut is performed as a cut between some source node s and some target node t, such that (i) each of the two nodes will be in a different sub-graph and (ii) the sum of weights of all removed edges will be as small as possible. Our system determines each Mincut using an implementation of the method by Goldberg and Tarjan (1986).[7]

```
function CLUSTER($G$)
    $clusters \leftarrow \emptyset$
    $E \leftarrow$ GETEDGES($G$)                          ▷ Step 1
    $e \leftarrow$ GETEDGEWITHLOWESTWEIGHT($E$)
    $s \leftarrow$ GETSOURCENODE($e$)
    $t \leftarrow$ GETTARGETNODE($e$)
    $G' \leftarrow$ MINCUT($G, s, t$)                     ▷ Step 2
    $\mathcal{C} \leftarrow$ GETCONNECTEDCOMPONENTS($G'$)
    for all $G_s \in \mathcal{C}$ do                      ▷ Step 3
        if SIZE($G_s$) $<= 2$ then
            $clusters \leftarrow clusters \cup G_s$
        else
            $clusters \leftarrow clusters \cup$ CLUSTER($G_s$)
        end if
    end for
    return $clusters$;
end function
```

Figure 2: Pseudo code of our clustering algorithm

As our goal is to induce clusters that correspond to pairs of similar predicates, we set a maximum number of two nodes per cluster as stopping criterion. Given an input graph G, our algorithm recursively applies Mincuts in three steps as described in Figure 2. Step 1 identifies the edge e with lowest weight in the given graph G. Step 2 performs the actual Mincut operation on G. Finally, the stopping criterion and recursion are applied in Step 3. An example of a clustered graph is illustrated in Figure 1.

The advantage of our method compared to off-the-shelf clustering techniques is two-fold: On the one hand, the clustering algorithm is free of any parameters, such as the number of clusters or a clustering threshold, that require fine-tuning. On the other hand, the approach makes use of a termination criterion that very well represents the nature of the goal of our task, namely to align pairs of predicates across comparable texts. The next section provides empirical evidence for the advantage of this approach.

5 Experiments

This section evaluates our graph-clustering model on the task of aligning predicates across comparable texts. For comparison to related tasks and methods, we describe different evaluation settings, vari-

[7] Basic graph operations are performed using the freely available Java library JGraph, cf. `http://jgrapht.org/`.

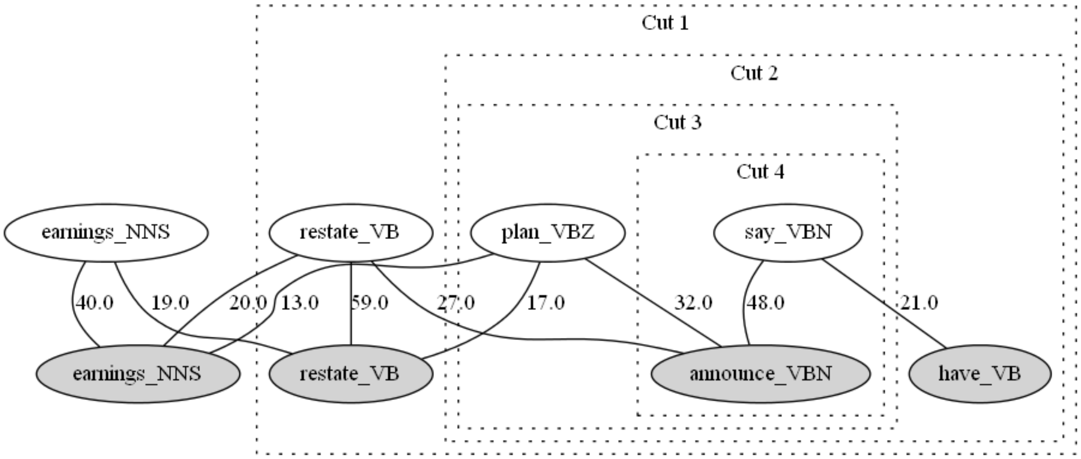

Figure 1: The predicates of two sentences (white: “The company has said it plans to restate its earnings for 2000 through 2002.”; grey: “The company had announced in January that it would have to restate earnings (. . .)”) from the Microsoft Research Paragraph Corpus are aligned by computing clusters with minimum cuts.

ous baselines, as well as results for these baselines and the model presented above.

5.1 Settings

In order to benchmark our model against traditional methods for word alignment, we first apply our graph-based alignment model (**Full**) on three sentence-based paraphrase corpora. This model uses the similarity measures defined in Section 4.2 and the clustering algorithm introduced in Section 4.3.

In a second experiment, we evaluate **Full** on our novel task of inducing predicate alignments across comparable monolingual texts, using the **GigaPairs** data set described in Section 3. We evaluate against the manually annotated gold alignments in the test data set described in Section 3.2. To gain more insight into the performance of the various similarity measures included in the **Full** model, we evaluate simplified versions that omit individual similarity measures (**Full–[measure name]**).

The relative differences in performance against various baselines will help us quantify the differences and difficulties between a traditional sentence-based word alignment setting and our novel alignment task that operates on full texts.

5.1.1 Sentence-level Alignment Setting

For sentence-based predicate alignment we make use of the following three corpora that are word-aligned subsets of the paraphrase collections described in (Cohn et al., 2008): **MTC** consists of 100 sentence pairs from the Multiple-Translation Chinese Corpus (Huang et al., 2002), **Leagues** contains 100 sentential paraphrases from two translations of Jules Verne’s “Twenty Thousand Leagues Under the Sea”, and **MSR** is a sub-set of the Microsoft Research Paraphrase Corpus (Dolan and Brockett, 2005), consisting of 130 sentence pairs. All three paraphrase collections are in English.

Results for these experiments are reported in Section 5.3.1. Note that in order to determine alignment candidates, we apply the same pre-processing steps as used for the annotation of our corpus. The semantic parser identified an average number of 3.8, 5.1 and 4.7 predicates per text (i.e., per paraphrase sentence) in **MTC**, **Leagues** and **MSR**, respectively. All models are evaluated against the subset of gold standard alignments (cf. Cohn et al. (2008)) between pairs of words marked as predicates.

5.1.2 Text-level Alignment Setting

Results for our own data set, **GigaPairs**, are reported in Section 5.3.2. In this setting, models are evaluated against the annotated gold standard alignments between predicates as described in Section 3.2. Since all text pairs in **GigaPairs** comprise multiple sentences each, the average number of predicates per text to consider (27.5) is much higher than in the paraphrase settings. As the full graph representation becomes rather inefficient to handle (by default, edges are inserted between all predicate pairs), we use the development set of 10 text pairs to estimate

	MTC			Leagues			MSR		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
LemmaId	25.1**	74.9	37.6**	31.5**	67.2	42.9**	42.3**	90.8	57.7**
Greedy	74.8**	**88.3****	81.0	75.0**	**86.0****	80.1	80.7**	**97.0****	88.1
WordAlign	**99.3**	86.6	**92.5**	**98.7**	78.5	**87.4**	**99.5**	96.0*	**97.7***
Full	92.3	72.2	81.1	92.7	69.4	79.4	94.5	88.3	91.3

Table 1: Results for sentence-based predicte alignment in the three benchmark settings **MTC**, **Leagues** and **MSR** (all numbers in %); results that significantly differ from **Full** are marked with asterisks (* $p<0.05$; ** $p<0.01$).

a threshold on predicate similarity for adding edges. We tested all thresholds from 1.5 to 4.0 with a stepsize of 0.25 and found 2.5 to perform best. This threshold is applied in the evaluation of all graph-based models.

5.2 Baselines

A simple baseline for both settings is to align all predicates whose lemmas are identical. This baseline, henceforth called **LemmaId**, is computed as a lower bound for all settings. In order to assess the benefits of the clustering step, we propose a second baseline that uses the same similarity measures and thresholds as our **Full** model, but omits the clustering step described in Section 4.3. Instead, it greedily computes as many 1-to-1 alignments as possible, starting from the highest similarity to the learned threshold (**Greedy**).

As a more sophisticated baseline, we make use of alignment tools commonly used in statistical machine translation (SMT). For the three sentence-based paraphrase settings **MTC**, **Leagues** and **MSR**, Cohn et al. (2008) readily provide GIZA++ (Och and Ney, 2003) alignments as part of their word-aligned paraphrase corpus. For the experiments in the **GigaPairs** setting, we train our own word alignment model using the state-of-the-art word alignment tool Berkeley Aligner (Liang et al., 2006). As word alignment tools require pairs of sentences as input, we first extract paraphrases in the latter setting using a re-implementation of the paraphrase detection system by Wan et al. (2006).[8] In the following section, we abbreviate both baselines using SMT alignment tools as **WordAlign**.

5.3 Results

We measure precision as the number of predicted alignments that are annotated in the gold standard divided by the total number of predictions. Recall is measured as the number of correctly predicted *sure* alignments divided by the total number of *sure* alignments in the gold standard. This conforms to evaluation measures used for word alignment models in SMT (Och and Ney, 2003). Following Cohn et al. (2008), we subsequently compute the F$_1$-score as the harmonic mean between precision and recall.

We compute statistical significance of result differences with a paired t-test (Cohen, 1995) over the affected test set documents and provide corresponding significance levels where appropriate.

5.3.1 Sentence-level Predicate Alignment

The results for **MTC**, **Leagues** and **MSR** are presented in Table 1. The numbers indicate that **WordAlign** consistently outperforms all other models on the three data sets in terms of F$_1$-score. Statistical significance of result differences between **WordAlign** and **Full** can only be observed for recall and F_1-score on the **MSR** data set ($p<0.05$). Other differences are not significant due to high variance of results compared to data set sizes.

The overall performance of **WordAlign** does not come much as a surprise, seeing that all three data sets consist of highly parallel sentence pairs. In fact, the results for **LemmaId** show that by aligning all predicates with identical lemmas, most of the sure alignments in the three settings are already covered. The reason for the low precision lies in the fact that the same lemma can occur multiple times in the same paraphrase, a phenomenon that is better handled by **WordAlign**, **Greedy** and **Full**. Interestingly, the **Greedy** model achieves the highest recall in all settings but it performs below our **Full**

[8] Note that the performance of this system lies slightly below the state-of-the-art results reported by Socher et al. (2011) However, we were not able to reproduce the results of Socher et al. using the publicly available release of their software.

model in terms of precision and F_1-score. The performance differences between **Greedy** and **Full** are statistically significant ($p<0.01$) regarding precision and recall.

5.3.2 Text-level Predicate Alignment

We now turn to the experiments on our own data set, **GigaPairs**, which comprises full documents of unequal lengths instead of pairs of single sentences. Table 2 presents the results for our full model and the three baselines. From all four approaches, **WordAlign** yields lowest performance. We observe two main reasons for this: On the one hand, sentence paraphrase detection does not perform perfectly. Hence, the extracted sentence pairs do not always contain gold alignments. On the other hand, even sentence pairs that contain gold alignments are generally less parallel than in the previous settings, which make them harder to align. The increased difficulty can also be seen in the results for the **Greedy** baseline, which only achieves an F_1-score of 20.1% in this setting. In contrast, we observe that the majority of all sure alignments (60.3%) can be retrieved by applying the **LemmaId** model.

The **Full** model achieves a recall of 46.6%, but it significantly outperforms **LemmaId** ($p<0.01$) in terms of precision (58.7%, +18.4 percentage points). This is an important factor for us, as we plan to use the alignments in subsequent tasks. With 52.0%, **Full** achieves the best overall F_1-score.

Ablating similarity measures. All aforementioned results were conducted in experiments with a uniform weighting scheme of similarity measures as introduced in Section 4.3. Table 3 shows the performance impact of individual similarity measures by removing them completely (i.e., setting their weight to 0.0). The numbers indicate that not all measures contribute positively to the overall performance when using equal weights. However, a significant difference can only be observed when removing the argument similarity measure, which drastically reduces the results. This clearly highlights the importance of incorporating the context of individual predications in this task.

Tuning weights. Subsequently, we tested various combinations of weights on our development set in order to estimate a good overall weighting scheme.

	Precision	Recall	F1
LemmaId	40.3**	**60.3****	48.3
Greedy	19.6**	20.6**	20.1**
WordAlign	19.7**	15.2**	17.2**
Full	**58.7**	46.6	**52.0**

Table 2: Results for **GigaPairs** (all numbers in %); results that significantly differ from **Full** are marked with asterisks (* $p<0.05$; ** $p<0.01$).

	Precision	Recall	F1
Full–WN	58.9	48.0	52.9
Full–VN	57.3	48.7	52.6
Full–Dist	54.3	42.8	47.9
Full–Args	40.1**	24.0**	30.0**
Full	58.7	46.6	52.0
Full+tuned	**59.7****	**50.7****	**54.8****

Table 3: Impact of removing individual measures and using a tuned weighting scheme (all numbers in %); results that significantly differ from **Full** are marked with asterisks (* $p<0.05$; ** $p<0.01$).

This tuning procedure is implemented as a brute-force technique, in which we fix the weight of one similarity measure and allow all other measures to receive a weight assignment between 0.25 to 5.0 times the fixed weight. Finally, the resulting weights are normalized to sum to 1.0. We found the best performing weighting scheme to be 0.09, 0.48, 0.24 and 0.19 for $\lambda_1, \ldots, \lambda_4$, respectively (cf. Eq. (2), Section 4). The performance gains of the resulting model (**Full+tuned**) can be seen in Table 3. Computing statistical significance of the result differences between **Full+tuned** and all baseline models confirmed significant improvements ($p<0.01$) for both precision and F_1-score.

5.4 Error Analysis

We perform an error analysis on the output of **Full+tuned** on the development set of **GigaPairs** in order to determine re-occurring problems. In total, the model missed 13 out of 35 sure alignments (Type I errors) and predicted 23 alignments not annotated in the gold standard (Type II errors).

Six Type I errors (46%) occurred when the lemma of an affected predicate occurred more than once in a text and the model missed a correct link. Vice versa, identical predicates that refer to different events have been the source of 8 Type II errors (35%). We observe that these errors are frequently related to predicates, such as “say” and “appear”, that often occur in news texts. Altogether, we find 15 Type II errors (65%) that are due to high predicate similarity despite low argument overlap (cf. Example (5)).

(5) a. The US alert (...) followed intelligence reports that ...
b. The Foreign Ministry announcement called on Japanese citizens to be cautious ...

We observe that argument overlap itself can be low even for correct alignments. This clearly indicates that a better integration of context is needed. Example (6.a) illustrates a case in which the agent of a warning event is not realized. Here, contextual information is required to correctly align it to the first warning event in (6.b). This involves inference beyond the local PAS.

(6) a. The US alert (...) is one step down from a full $[\text{travel}]_{\text{Arg1}}$ warning $[\ ]_{\text{Arg0}}$.
b. Japan has issued a travel alert ...(which) follows similar warnings $[\text{from American and British authorities}]_{\text{Arg0}}$. (...) An official said it was highly unusual for $[\text{Tokyo}]_{\text{Arg0}}$ to issue such a warning ...

6 Conclusion

We presented a novel task for *predicate alignment* across comparable monolingual texts, which we address using graph-based clustering with Mincuts. The motivation for this task is to acquire empirical data for studying discourse coherence factors related to argument structure realization.

As a first step, we constructed a data set of comparable texts that provide full discourse contexts for alternative verbalizations of the same underlying events. The data set is derived from all newswire pairs found in the English Gigaword Fifth Edition and contains a total of more than 160,000 paired documents.

A subset of these pairs forms an evaluation set, annotated with gold alignments that relate predications, which exhibit a (possibly partial) corresponding argument structure. We established that the annotation task, while difficult, can be performed with good inter-annotator agreement (κ at 0.86).

Our main contribution is a novel clustering approach using Mincuts for aligning predications across comparable texts. Our experiments established that recursive clustering improves on greedy selection methods by profiting from global information encoded in the graph representation. While the Mincut-based method is in itself unsupervised, a small amount of development data is needed to tune parameters for the construction of particularly suitable input graphs.

We tested our full model against two additional baselines: simple heuristic alignment based on identical lemma forms and a combination of techniques from SMT and paraphrase detection. The evaluation for our novel task was complemented by a traditional word alignment task using established paraphrase data sets. We determined clear differences in performance for all models for the two types of task settings. While word alignment methods from SMT outperform the competing models in the sentence-based alignment tasks, they perform poorly in the discourse setting.

In future work, we will enhance our model by incorporating more refined similarity measures including discourse-based criteria. We will further explore tuning techniques, e.g., a more suitable preselection method for edges in graph construction, in order to increase either precision or recall. The decision of optimizing towards one measure or another is clearly task-dependent. In our case, high precision is favorable as we plan to learn accurate discourse model parameters from the computed alignments. Even though such an optimization will result in an overall lower recall, application of the alignment model on the entire GigaPairs corpus can still provide us with a large amount of precise predicate alignments. Using this set of alignments, we will then proceed to exploit contextual information in order to learn a semantic model for discourse coherence in argument structure realization.

Acknowledgements

We are grateful to the Landesgraduiertenförderung Baden-Württemberg for funding within the research initiative “Coherence in language processing” at Heidelberg University. We thank Danny Rehl and Lukas Funk for annotation.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 Task 6: A pilot on semantic textual similarity. In *Proceedings of the 6th International Workshop on Semantic Evaluations*, Montreal, Canada, June. to appear.

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Regina Barzilay and Lillian Lee. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics,* Boston, Mass., 2–7 May 2004, pages 113–120.

Anja Belz, Eric Kow, Jette Viethen, and Albert Gatt. 2009. The grec main subject reference generation challenge 2009: overview and evaluation results. In *Proceedings of the 2009 Workshop on Language Generation and Summarisation*, pages 79–87.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth pascal recognizing textual entailment challenge. In *Proceedings of TAC*.

Anders Björkelund, Bernd Bohnet, Love Hafdell, and Pierre Nugues. 2010. A high-performance syntactic and semantic dependency parser. In *Coling 2010: Demonstration Volume*, pages 33–36, Beijing, China, August. Coling 2010 Organizing Committee.

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China, August. Coling 2010 Organizing Committee.

Chris Brockett. 2007. *Aligning the RTE 2006 Corpus*. Microsoft Research.

Peter F. Brown, Vincent J. Della Pietra, Stephan A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19:263–311.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.

Paul R. Cohen. 1995. *Empirical methods for artificial intelligence*. MIT Press, Cambridge, MA, USA.

Trevor Cohn, Chris Callison-Burch, and Mirella Lapata. 2008. Constructing Corpora for Development and Evaluation of Paraphrase Systems. 34(4).

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In J. Quiñonero-Candela, I. Dagan, and B. Magnini, editors, *Machine Learning Challenges*, pages 177–190. Springer, Heidelberg, Germany.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing*.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, Mass.

Adrew V. Goldberg and Robert E. Tarjan. 1986. A new approach to the maximum flow problem. In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 136–146, New York, NY, USA.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Weiwei Guo and Mona Diab. 2011. Semantic topic models: Combining word distributional statistics and dictionary definitions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 552–561, July.

Shudong Huang, David Graff, and George Doddington. 2002. *Multiple-Translation Chinese Corpus*. Linguistic Data Consortium, Philadelphia.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. A Large-scale Classification of English Verbs. 42(1):21–40.

Thomas K. Landauer and Susan T. Dumais. 1997. A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104:211–240.

Percy Liang, Benjamin Taskar, and Dan Klein. 2006. Alignment by agreement. In *North American Association for Computational Linguistics (NAACL)*, pages 104–111.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning,* Madison, Wisc., 24–27 July 1998, pages 296–304.

Bill MacCartney, Michael Galley, and Christopher D. Manning. 2008. A phrase-based alignment model for natural language inference. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing,* Waikiki, Honolulu, Hawaii, 25-27 October 2008.

Adam Meyers, Ruth Reeves, and Catherine Macleod. 2008. *NomBank v1.0*. Linguistic Data Consortium, Philadelphia.

Shachar Mirkin, Jonathan Berant, Ido Dagan, and Eyal Shnarch. 2010a. Recognising entailment within discourse. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, Beijing, China, August. Coling 2010 Organizing Committee.

Shachar Mirkin, Ido Dagan, and Sebastian Padó. 2010b. Assessing the role of discourse references in entailment inference. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics,* Uppsala, Sweden, 11–16 July 2010.

Jeff Mitchell and Mirella Lapata. 2010. Composition in Distributional Models of Semantics. 34(8):1388–1429.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. 29(1):19–51.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–105.

Robert Parker, David Graff, Jumbo Kong, Ke Chen, and Kazuaki Maeda. 2011. *English Gigaword Fifth Edition*. Linguistic Data Consortium, Philadelphia.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. WordNet::Similarity – Measuring the relatedness of concepts. In *Companion Volume to the Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics,* Boston, Mass., 2–7 May 2004, pages 267–270.

Tom Richens. 2008. Anomalies in the wordnet verb hierarchy. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 729–736. Association for Computational Linguistics.

Michael Roth and Anette Frank. 2012. Aligning predicate argument structures in monolingual comparable texts: A new corpus for a new task. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics*, Montreal, Canada, June.

Josef Ruppenhofer, Caroline Sporleder, Roser Morante, Collin Baker, and Martha Palmer. 2010. SemEval-2010 Task 10: Linking Events and Their Participants in Discourse. In *Proceedings of the 5th International Workshop on Semantic Evaluations*, pages 45–50, Uppsala, Sweden, July.

Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Advances in Neural Information Processing Systems (NIPS 2011)*.

Ivan Titov and Mikhail Kozhevnikov. 2010. Bootstrapping semantic analyzers from non-contradictory texts. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics,* Uppsala, Sweden, 11–16 July 2010, pages 958–967.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. *ACE 2005 Multilingual Training Corpus*. Linguistic Data Consortium, Philadelphia.

Stephen Wan, Mark Dras, Robert Dale, and Cecile Paris. 2006. Using dependency-based features to take the ”Para-farce” out of paraphrase. In *Proceedings of the Australasian Language Technology Workshop*, pages 131–138.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2011. *OntoNotes Release 4.0*. Linguistic Data Consortium, Philadelphia.

Sander Wubben, Antal van den Bosch, Emiel Krahmer, and Erwin Marsi. 2009. Clustering and matching headlines for automatic paraphrase acquisition. In *Proceedings of the 12th European Workshop on Natural Language Generation (ENLG 2009)*, pages 122–125, Athens, Greece, March. Association for Computational Linguistics.