Predicate Argument Alignment using a Global Coherence Model

Travis Wolfe, Mark Dredze, and Benjamin Van Durme
Johns Hopkins University
Baltimore, MD, USA

Abstract
We present a joint model for predicate argument alignment. We leverage multiple sources of semantic information, including temporal ordering constraints between events. These are combined in a max-margin framework to find a globally consistent view of entities and events across multiple documents, which leads to improvements over a very strong local baseline.

1 Introduction

Natural language understanding (NLU) requires analysis beyond the sentence-level. For example, an entity may be mentioned multiple times in a discourse, participating in various events, where each event may itself be referenced elsewhere in the text. Traditionally the task of coreference resolution has been defined as finding those entity mentions within a single document that co-refer, while cross-document coreference resolution considers a wider discourse context across many documents, yet still pertains strictly to entities.

Predicate argument alignment, or entity-event cross-document coreference resolution, enlarges the set of possible co-referent elements to include the mentions of situations in which entities participate. This expanded definition drives practitioners towards a more complete model of NLU, where systems must not only consider who is mentioned, but also what happened. However, despite the drive towards an expanded notion of discourse, models typically are formulated with strong notions of local-independence: viewing a multi-document task as one limited to individual pairs of sentences. This creates a mis-match between the goals of such work – considering entire documents – with the systems – consider individual sentences.

In this work, we consider a system that takes a document level view in considering coreference for entities and predictions: the task of predicate argument linking. We treat this task as a global inference problem, leveraging multiple sources of semantic information identified at the document level. Global inference for this problem is mostly unexplored, with the exception of Lee et al. (2012) (discussed in §8). Especially novel here is the use of document-level temporal constraints on events, representing a next step forward on the path to full understanding.

Our approach avoids the pitfalls of local inference while still remaining fast and exact. We use the pairwise features of a very strong predicate argument aligner (Wolfe et al., 2013) (competitive with the state-of-the-art (Roth, 2014)), and add quadratic factors that constrain local decisions based on global document information. These global factors lead to superior performance compared to the previous state-of-the-art. We release both our code and data.1

2 Model

Consider the two sentences from the document pair shown in Figure 1. These sentences describe the same event, although with different details. The source sentence has four predicates and four arguments, while the target has three predicates and three arguments. In this case, one of the predicates from each sentence aligns, as do three of the arguments. We also show additional information potentially helpful to determining alignments: temporal relations between the predicates. The goal of predicate argument alignment is to assign these links indicating coreferent predicates and arguments across a document pair (Roth and Frank, 2012).

Previous work by Wolfe et al. (2013) formulated

1https://github.com/hltcoe/parma2
Figure 1: An example analysis and predicate argument alignment task between a source and target document. Predicates appear as hollow ovals, have blue mentions, and are aligned considering their arguments (dashed lines). Arguments, in black diamonds with green mentions, represent a document-level entity (coreference chain), and are aligned using their predicate structure and mention-level features. The alignment choices appear in the middle in red. Temporal relation information is lifted into the global inference over alignments.

We present a global predicate argument alignment model based on considering quadratic interactions between alignment variables to captures patterns we expect in coherent discourse. We introduce factors which are comprised of a binary variable, multiple quadratic constraints on that variable, and features that determine the cost associated with that variable in order to characterize the dependence between alignment decisions.

While the mathematical framework we use is similar to Lacoste-Julien et al. (2006), predicate argument alignment greatly differs from word alignment; thus our joint factors are based on different sources of regularity. Word alignment favors monotonicity in word order, but this effect is very weak in predicate argument alignment: aligned items can be spread throughout a document, and are often nested, gapped, or shuffled. Instead, we encode assumptions about consistency of temporal relations between coreferent events, coherence between predicates and arguments that appear in both documents, and fertility (to prevent over-alignment). We also note that our setting has much less data than typical word alignment tasks, as well as richer features that utilize semantic resources.
Notation An alignment between an item indexed by $i$ in the source document and $j$ in the target document is represented by variable $z_{ij} \in \{0, 1\}$, where $z_{ij} = 1$ indicates that items $i$ and $j$ are aligned. In some cases, we will explicitly indicate when the two items are predicates as $z_{ij}^p$; an argument alignment will be $z_{ij}^a$. We represent all alignments for a document pair as matrix $z$.

For clarity, we omit any variable representing observed data when discussing feature functions; alignment variables are endowed with this information. For each pair of items we use “local” feature functions $f(\cdot)$ and corresponding parameters $w$, which capture the similarity between two items without the context of other alignments.

$$s_{ij} = w \cdot f(z_{ij}) \quad (1)$$

where $s_{ij}$ is the score of linking items $i$ and $j$.

Using only local features, our system would greedily select alignments. To capture global aspects we add joint factors that capture effects between alignment variables. Each joint factor $\phi$ is comprised of a constrained binary variable $z_\phi$ associated with features $f(\cdot)$ that indicates when the factor is active. Together with parameters $w$ these form additional scores $s_\phi$ for the objective:

$$s_\phi = w \cdot f(\phi) \quad (2)$$

The full linear scoring function on alignments sums over both local similarity and joint factors:

$$\sum_{ij} s_{ij} z_{ij} + \sum_{\phi \in \Phi} s_\phi z_\phi. \quad (3)$$

Lastly, it is convenient to describe the local feature functions and their corresponding alignment variable as factors with no constraints, and we will do so when describing the full score function.

3 Local Factors

Local factors encode features based on the mention pair, which include a wide variety of similarity measures, e.g. whether two headwords appear as synonyms in WordNet, gender agreement based on possessive pronouns. We adopt the features of Wolfe et al. (2013), a strong baseline system which doesn’t use global inference. These features are built on top of a variety of semantic resources (PPDB (Ganitkevitch et al., 2013), WordNet (Miller, 1995), FrameNet (Baker et al., 1998)) and methods for comparing mentions (tree edit distance (Yao et al., 2013), string transducer (Andrews et al., 2012)).

4 Joint Factors

Our goal is to develop joint factors that improve over the feature rich local factors baseline by considering global information.

Fertility A common mistake when making independent classification decisions is to align many source items to a single target item. While each link looks promising on its own, they clearly cannot all be right. Empirically, the training set reveals that many to one alignments are uncommon; thus many to one predictions are likely errors. We add a fertility factor for predicates and arguments, where fertility is defined as the number of links to an item. Higher fertilities are undesired and are thus penalized. Formally, for matrix $z$, the fertility of a row $i$ or column $j$ is the sum of that row or column. We discuss fertility in terms of rows below.

We include two types of fertility factors. First, factor $\phi_{\text{fert}1}$ distinguishes between rows with at least one link from those with none. For row $i$, we add one instance of the linear factor $\phi_{\text{fert}1}$ with constraints

$$z_{\phi_{\text{fert}1}} \geq z_{ij} \quad \forall j \quad (4)$$

The cost associated with $z_{\phi_{\text{fert}1}}$, which we will refer to as $s_{\text{fert}1}$, will be incurred any time an item is mentioned in both documents. For data sets with many singletons, $s_{\text{fert}1}$ more strongly penalizes non-singleton rows, reflecting this pattern in the training data. We make $s_{\text{fert}1}$ parametric, where the features of the $\phi_{\text{fert}1}$ factor allow us to learn different weights for predicates and arguments, as well as the size of the row, i.e. number of items in the pairing.

The second fertility factory $\phi_{\text{fert}2}$ considers items with a fertility greater than one, penalizing items for having too many links. Its binary variable has the

$$s_{\phi_{\text{fert}2}} \geq z_{ij} \quad \forall j \quad (5)$$

Some features inspect the apparent predicate argument structure, based on things like dependency parses, but the model may not inspect more than one of its own decisions (joint factors) while scoring an alignment.
within our dataset (2006). We parameterize fertility for only two cases compared with Lacoste-Julien et al. To constrain the variable of \( \phi_{psa} \), we add a quadratic constraint that considers every possible pair of argument alignments that might overlap:

\[
\phi_{psa} \geq \sum_{ij} z_{ij} \left( 1 - \max_k \frac{z_{kl}}{\max_l z_{kl}} \right)
\]

where \( \text{args}(p_i) \) finds the indices of all arguments governed by the predicate \( p_i \).

**Entity-centric** We expect similar behavior from arguments (entities). If an entity appears in two documents, it is likely that this entity will be mentioned in the context of a common predicate, i.e. arguments share predicates (\( \text{asp} \)). For a given argument alignment \( z_{ij}^p \) we add quadratic constraints so that \( \phi_{asp} \) represents a penalty for two arguments not sharing a single predicate:

\[
\phi_{asp} \geq \sum_{ij} \left( 1 - \frac{\max_{k \in \text{args}(p_i)} z_{kl}^p}{\max_{l \in \text{args}(p_j)} z_{kl}^p} \right)
\]

where \( \text{preds}(a_i) \) finds the indices of all predicates that govern any mention of argument \( a_i \).

The features \( f(\phi) \) for both \( \phi_{psa} \) and \( \phi_{asp} \) are an intercept feature and a bucketed count of the size of \( \text{args}(p_i) \times \text{args}(p_j) \) or \( \text{preds}(a_i) \times \text{preds}(a_j) \) respectively.

**Temporal Information** Temporal ordering, in contrast to textual ordering, can indicate when predicates cannot align: we expect aligned predicates in both documents to share the same temporal relations. SemEval 2013 included a task on predicting temporal relations between events (UzZaman et al., 2013). Many systems produced partial relations of events in a document based on lexical aspect and tense, as well as discourse connectives like “during” or “after”. We obtain temporal relations with CAEVO, a state-of-the-art sieve-based system (Chambers et al., 2014).
In-Doc Relations allowing feature values of at most $\epsilon$-abilities are not perfectly calibrated. We use $\epsilon$-binary variables. There is a local similarity score $Z$. The objective is a linear function over two relations are incompatible, e.g. \textit{spouseOf} for incorporating relational information into coreference matches one of the conflict patterns above, we add a factor using $z_{\phi_{\text{emp}}}$:

$$z_{\phi_{\text{emp}}} \geq z_{\phi_{\text{emp}}} \cdot 2 \text{bx}$$

if $p_a R_1 p_b, p_x R_2 p_y, R_1 = R_2$

$$z_{\phi_{\text{emp}}} \geq z_{\phi_{\text{emp}}} \cdot 2by$$

if $p_a R_1 p_b, p_x R_2 p_y, R_1 = R_2^{-1}$

(8)

Thus $s_{\phi_{\text{emp}}}$ is the cost of disagreeing with the in-doc temporal relations. This is a general technique for incorporating relational information into coreference decisions. It only requires specifying when two relations are incompatible, e.g. \textit{spouseOf} and \textit{siblingOf} are incompatible relations (in most states). We leave this for future work.

Since CAEVO gives each relation prediction a probability, we incorporate this into the feature by indicating the probability of a conflict not arising:

$$f(\phi_{\text{emp}}) = \log (1 - p(R_1)p(R_2) + \epsilon)$$

(9)

$\epsilon$ avoids large negative values since CAEVO probabilities are not perfectly calibrated. We use $\epsilon = 0.1$, allowing feature values of at most $-2.3$.

**Summary** The objective is a linear function over binary variables. There is a local similarity score coefficient on every alignment variable, and a joint factor similarity score on every quadratic variable. These quadratic variables are constrained by products of the original alignment variables. Decoding an alignment requires solving this quadratically constrained integer program; in practice is can be solved quickly without relations.

5 Inference

**Learning** We use the supervised structured SVM formulation of Joachims et al. (2009). As is common in structure prediction we use margin rescaling and slack variable, with the structural SVM objective:

$$\min_{w} \|w\|^2_2 + C\xi$$

s.t. $\xi \geq 0$

$$\xi + \sum_{i=1}^{N} w \cdot f(z_i) \geq \sum_{i=1}^{N} w \cdot f(\hat{z}_i) + \Delta(z_i, \hat{z}_i)$$

$\forall \hat{z}_i \in Z_i$

(10)

where $Z_i$ is the set of all possible alignments that have the same shape as $z_i$.
The score function for an alignment uses three types of terms: weights, features, and alignment variables. When we decode, we take the product of the weights and the features to get the costs for the ILP (e.g. $s_{ij} = w \cdot f(\phi)$). When we optimize our SVM objective, we take the product of the alignment variables and the features to get modified features for the SVM:

$$f(z) = \sum_{ij} z_{ij} f(z_{ij}) + \sum_{\phi \in \Theta} z_{\phi} f(\phi) \quad (11)$$

Since we cannot iterate over the exponentially many margin constraints, we solve for this optimization using the cutting-plane learning algorithm. This algorithm repeatedly asks the “separation oracle” for the most violated SVM constraint, which finds this constraint by solving:

$$\arg \max_{\tilde{z}_1, \ldots, \tilde{z}_N} \sum_{i} w \cdot f(\tilde{z}_i) + \Delta(\tilde{z}_i, \tilde{z}_i) \quad (12)$$

subject to the constraints defined by the joint factors. When the separation oracle returns a constraint that is not violated or is already in the working set, then we have a guarantee that we solved the original SVM problem with exponentially many constraints. This is the most time-consuming aspect of learning, but since the problem decomposes over document alignments, we cache solutions on a per document alignment basis. With caching, we only call the separation oracle around 100-300 times.

We implement the separation oracle using an ILP solver, CPLEX, due to complexity of the discrete optimization problem: there are $2^m$ possible alignments for and $m \times n$ alignment grid. In practice this is solved very efficiently, taking less than a third of a second per document alignment on average. We would like $\Delta$ to be F1, but we need a decomposable loss to include it in a linear objective (Taskar et al., 2003). Instead, we use Hamming loss as a surrogate, as in Lacoste-Julien et al. (2006).

Our training data is heavily biased towards negative examples, performing poorly on F1 since precision and recall are unbalanced. We use an asymmetric version of Hamming loss that incurs $c_{FP}$ cost for predicting an alignment for two unaligned items and $c_{FN}$ for predicting no alignment for two aligned items. We fixed $c_{FP} = 1$ and tuned $c_{FN} \in \{1, 2, 3, 4\}$ on dev data. Additionally we found it useful to tune the scale of the loss function across $\{1/2, 1, 2\}$. Previous work, such as Joachims et al. (2009), use a hand-chosen constant for the scale of the Hamming loss, but we observe some sensitivity in this parameter and choose to optimize it.

Decoding Following Wolfe et al. (2013), we tune the threshold for classification $\tau$ on dev data to maximize F1 (via linesearch). For SVMs $\tau$ is typically fixed at 0: this is not necessarily good practice when your training loss differs from test loss (Hamming vs F1). In our case this extra parameter is worth allocating a portion of training data to enable tuning. Tuning $\tau$ addresses the same problem as using an asymmetric Hamming loss, but we found that doing both led to better results. Since we are using a global scoring function rather than a set of classifications, $\tau$ is implemented as a test-time unary factor on every alignment.

6 Experiments

Data We consider two datasets for evaluation. The first is a cross-document entity and event coreference resolution dataset called the Extended Event Coref Bank (EECB) created by Lee et al. (2012) and based on a corpus from Bejan and Harabagiu (2010). The dataset contains clusters of news articles taken from Google News with annotations about coreference entities and events. Following the procedure of Wolfe et al. (2013), we select the first document in every cluster and pair it with every other document in the cluster.

The second dataset (RF) comes from Roth and Frank (2012). The dataset contains pairs of news articles that describe the same news story, and are annotated for predicate links between the document pairs. Due to the lack of annotated arguments, we can only report predicate linking performance and the psa and asp factors do not apply. Lastly, the size of the RF data should be noted as it is much smaller than EECB: the test set has 60 document pairs and the dev set has 10 document pairs.

\footnote{Only tuning $\tau$ performed almost as well as tuning $\tau$ and the Hamming loss, but not tuning $\tau$ performed much worse than only tuning the Hamming loss at train time.}
Both datasets are annotated with parses and in-document coreference labels provided by the toolset of Napoles et al. (2012) and are available with our code release. Due to the small data size, we use $k$-fold cross validation for both datasets. We choose $k = 10$ for RF due to its very small size (more folds give more training examples) and $k = 5$ on EECB to save computation time (amount of training data in EECB is less of a concern). Hyperparameters were chosen by hand using cross validation on the EECB dataset using F1 as the criteria (rather than Hamming). Figures report averages across these folds.

**Systems** Following Roth and Frank (2012) and Wolfe et al. (2013) we include a *Lemma* baseline for identifying alignments which will align any two predicates or arguments that have the same lemmatized head word. The *Local* baseline uses the same features as Wolfe et al., but none of our joint factors. In addition to running our joint model with all factors, we measure the efficacy of each individual factor by evaluating each with the local features.

For evaluation we use a generous version of F1 that is defined for alignment labels composed of sure, $G_s$, and possible links, $G_p$ and the system’s proposed links $H$ (following Cohn et al. (2008), Roth and Frank (2012) and Wolfe et al. (2013)).

$$P = \frac{|H \cap G_p|}{|H|}, \quad R = \frac{|H \cap G_s|}{|G_s|}, \quad F = 2\frac{PR}{P + R}$$

Note that the EECB data does not have a sure and possible distinction, so $G_s = G_p$, resulting in standard F1. In addition to F1, we separately measure predicate and argument F1 to demonstrate where our model makes the largest improvements.

We performed a one-sided paired-bootstrap test where the null hypothesis was that the joint model was no better than the *Local* baseline (described in Koehn (2004)). Cases where $p < 0.05$ are bolded.

### 7 Results

Results for EECB and RF are reported in Table 7. As previously reported, using just local factors (features on pairs) improves over lemma baselines (Wolfe et al., 2013). The joint factors make statistically significant gains over local factors in almost all experiments. Fertility factors provide the largest improvements from any single constraint. A fertility penalty actually allows the pairwise weights to be more optimistic in that they can predict more alignments for reasonable pairs, allowing the fertility penalty to ensure only the best is chosen. This penalty also prevents the “garbage collecting” effect that arises for instances that have rare features (Brown et al., 1993).

Temporal constraints are relatively sparse, appearing just 2.8 times on average. Nevertheless, it was very helpful across all experiments, though only statistically significantly on the RF dataset. This is one of the first results to demonstrate benefits of temporal relations affecting an downstream task. Perhaps surprisingly, these improvements result from a a temporal relation system that has relatively poor absolute performance. Despite this, improvements are possibly due to the orthogonal nature of temporal information; no other feature captures this signal. This suggests that future work on temporal relation prediction may yield further improvements and deserves more attention as a useful feature for semantic tasks in NLP.

The predicate-centric factors improved performance significantly on both datasets. For the predicate-centric factor, when a predicate was aligned there is a 72.3% chance that there was at least one argument aligned as well, compared to only 14.1% of case of non-aligned predicates. As mentioned before, the reason the former number isn’t 100% is primarily due to implicit arguments and errors in argument identification. The argument-centric features helped almost as much as the predicate-centric version, but the improvements were not significant on the EECB dataset. Running the same diagnostic as the predicate-centric feature reveals similar support: in 57.1% of the cases where an argument was aligned, at least one predicate it partook in was aligned too, compared to 7.6% of cases for non-aligned arguments. Both the...

---

5 https://github.com/cnap/anno-pipeline

6 The lemma baseline is obviously sensitive to the lemmatizer used. We used the Stanford CoreNLP lemmatizer (Manning et al., 2014) and found it yielded slightly better results than previously reported as the lemma baseline (Roth and Frank, 2012), so we used it for all systems to ensure fairness and that the baseline is as strong as it could be.
<table>
<thead>
<tr>
<th></th>
<th>F1</th>
<th>P</th>
<th>R</th>
<th>Arg F1</th>
<th>Arg P</th>
<th>Arg R</th>
<th>Pred F1</th>
<th>Pred P</th>
<th>Pred R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Lemma</strong></td>
<td>68.1</td>
<td>79.3</td>
<td>*59.6</td>
<td>61.7</td>
<td>79.1</td>
<td>*50.6</td>
<td>75.0</td>
<td>87.3</td>
<td>65.7</td>
</tr>
<tr>
<td><strong>Local</strong></td>
<td>73.0</td>
<td>75.8</td>
<td><strong>70.5</strong></td>
<td>67.7</td>
<td>76.3</td>
<td><strong>60.8</strong></td>
<td>78.7</td>
<td>81.4</td>
<td>76.2</td>
</tr>
<tr>
<td>+Fertility</td>
<td>77.1</td>
<td>83.9</td>
<td>*71.3</td>
<td>66.6</td>
<td>80.9</td>
<td>*56.6</td>
<td>82.8</td>
<td>87.4</td>
<td><strong>78.7</strong></td>
</tr>
<tr>
<td>+Predicate-centric</td>
<td>74.1</td>
<td>80.7</td>
<td>68.6</td>
<td>67.4</td>
<td>81.6</td>
<td>*57.3</td>
<td>79.7</td>
<td>85.0</td>
<td>75.1</td>
</tr>
<tr>
<td>+Argument-centric</td>
<td>73.7</td>
<td>81.2</td>
<td>67.5</td>
<td>66.8</td>
<td><strong>83.0</strong></td>
<td>*55.9</td>
<td>79.3</td>
<td>85.1</td>
<td>74.3</td>
</tr>
<tr>
<td>+Temporal</td>
<td>73.7</td>
<td>78.2</td>
<td>*69.7</td>
<td><strong>67.9</strong></td>
<td>80.6</td>
<td>58.7</td>
<td>79.0</td>
<td>82.1</td>
<td>76.1</td>
</tr>
<tr>
<td>+All Factors</td>
<td><strong>77.5</strong></td>
<td>*86.3</td>
<td>*70.3</td>
<td>65.8</td>
<td><strong>83.1</strong></td>
<td>*54.5</td>
<td><strong>83.7</strong></td>
<td><strong>89.7</strong></td>
<td>78.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Pred F1</th>
<th>Pred P</th>
<th>Pred R</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Lemma</strong></td>
<td>52.4</td>
<td>47.6</td>
<td>58.2</td>
</tr>
<tr>
<td><strong>Local</strong></td>
<td>58.1</td>
<td><strong>63.5</strong></td>
<td>53.6</td>
</tr>
<tr>
<td>+Fertility</td>
<td><strong>60.0</strong></td>
<td>57.4</td>
<td><strong>62.4</strong></td>
</tr>
<tr>
<td>+Predicate-centric</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>+Argument-centric</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>+Temporal</td>
<td>59.0</td>
<td>57.4</td>
<td>60.6</td>
</tr>
<tr>
<td>+All factors</td>
<td>59.4</td>
<td>56.9</td>
<td><strong>62.2</strong></td>
</tr>
</tbody>
</table>

Figure 3: Cross validation results for EECB (above) (Lee et al., 2012) and RF (left) (Roth and Frank, 2012). Statistically significant improvements from Local marked * ($p < 0.05$ using a one-sided paired-bootstrap test) and best results are bolded.

Predicate- and argument-centric improve similarly across both predicates and arguments on EECB.

While each of the joint factors all improve over the baselines on RF, the full model with all the joint factors does not perform as well as with some factors excluded. Specifically, the fertility model performs the best. We attribute this small gap to lack of training data (RF only contains 64 training document pairs in our experiments), as this is not a problem on the larger EECB dataset.

Additionally, the joint models seem to trade precision for recall on the RF dataset compared to the Local baseline. Note that both models are tuned to maximize F1, so this tells you more about the shape of the ROC curve as opposed to either models’ ability to achieve either high precision or recall. Since we don’t see this behavior on the EECB corpus, it is more likely that this is a property of the data than the model.

8 Related Work

The task of predicate argument linking was introduced by Roth and Frank (2012), who used a graph parameterized by a small number of semantic features to express similarities between predicates and used min-cuts to produce an alignment. This was followed by Wolfe et al. (2013), who gave a locally-independent, feature-rich log-linear model that utilized many lexical semantic resources, similar to the sort employed in RTE challenges.

Lee et al. (2012) considered a similar problem but sought to produce clusters of entities and events rather than an alignment between two documents with the goal of improving coreference resolution. They used features which consider previous event and entity coreference decisions to make future coreference decisions in a greedy manner. This differs from our model which is built on non-greedy joint inference, but much of the signal indicating when two mentions corefer or are aligned is similar.

In the context of in-document coreference resolution, Recasens et al. (2013) sought to overcome the problem of opaque mentions by finding high-precision paraphrases of entities by pivoting off verbs mentioned in similar documents. We address the issue of opaque mentions not by building a paraphrase table, but by jointly reasoning about entities that participate in coreferent events (c.f. §4); the approaches are complementary.

In this work we incorporate ordering information of events. Though we consider it an upstream task, there is a line of work trying to predict temporal relations between events (Pustejovsky et al., 2003; Mani et al., 2006; Chambers et al., 2014). Our results indicate this is a useful source of information, one of the first results to show an improvement from this

---

7 A lexically disparate description of an entity.
We utilize an ILP to improve upon a pipelined system, similar to Roth and Yih (2004), but our work differs in that we do not use piecewise-trained classifiers. Our local similarity scores are calibrated according to a global objective by propagating the gradient back from the loss to every parameter in the model. When using piecewise training, local classifiers must focus more on recall (in the spirit of Weiss and Taskar (2010)) than they would for an ordinary classification task with no global objective. Our method trains classifiers jointly with a global convex objective. While our training procedure requires decoding an integer program, the parameters we learn are globally optimal.

9 Conclusion

We presented a max-margin quadratic cost model for predicate argument alignment, seeking to exploit discourse level semantic features to improve on previous, locally independent approaches. Our model includes factors that consider fertility of predicates and arguments, the predicate argument structure present in coherent discourses, and soft constraints on predicate coreference determined by a temporal relation classifier. We have shown that this model significantly improves upon prior work which uses extensive lexical resources but without the benefit of joint inference. Additionally, this is one of the first demonstrations of the benefits of temporal relation identification. Overall, this work demonstrates the benefits of considering global document information as part of natural language understanding.

Future work should extend the problem formulation of predicate argument alignment to consider incremental linking: starting with a pair of documents, perform linking, and then continue to add in documents over time. This problem formulation would capture the evolution of a breaking news story, which closely matches the type of data (news articles) considered in this work (EECB and RF datasets). This formulation ties into existing work on news summarization, topic detection and tracking, and multi-document NLU. This goes hand with work on better intra-document relation prediction methods, such as the temporal relation model used in this work, to lead to better joint linking decisions.

References


Simon Lacoste-Julien, Benjamin Taskar, Dan Klein, and Michael I. Jordan. 2006. Word alignment via


