# Cross-Sentence Inference for Process Knowledge

**Samuel Louvan[+], Chetan Naik[+], Sadhana Kumaravel[+], Heeyoung Kwon[+], Niranjan Balasubramanian[+], Peter Clark[*]**

[+]Stony Brook University, [*]Allen Institute for AI,
{slouvan, cnaik, skumaravel, heekwon,niranjan}@cs.stonybrook.edu,
peterc@allenai.org

## Abstract

For AI systems to reason about real world situations, they need to recognize which processes are at play and which entities play key roles in them. Our goal is to extract this kind of role-based knowledge about processes, from multiple sentence-level descriptions. This knowledge is hard to acquire; while semantic role labeling (SRL) systems can extract sentence level role information about individual mentions of a process, their results are often noisy and they do not attempt create a globally consistent characterization of a process.

To overcome this, we extend standard *within sentence* joint inference to inference across multiple sentences. This *cross sentence* inference promotes role assignments that are compatible across different descriptions of the same process. When formulated as an Integer Linear Program, this leads to improvements over within-sentence inference by nearly 3% in F1. The resulting role-based knowledge is of high quality (with a F1 of nearly 82).

## 1 Introduction

Knowledge about processes is essential for AI systems in order to understand and reason about the world. At the simplest level, even knowing which class of entities play key roles can be useful for tasks involving recognition and reasoning about processes. For instance, given a description "a puddle drying in the sun", one can recognize this as an instance of the process *evaporation* using a macro-level role knowledge: Among others, the typical *undergoer* of evaporation is a kind of liquid (the puddle), and the *enabler* is usually a heat source (the sun).

| |
|---|
| 1) Evaporation is the process by which liquids are converted to their gaseous forms. |
| 2) Evaporation is the process by which water is converted into water vapor. |
| 3) Water vapor rises from water due to evaporation. |
| 4) Clouds arise as water evaporates in the sun. |

**Table 1:** Example sentences for the process *evaporation*. Underlined spans correspond to fillers for the *undergoer* role.

Our goal is to acquire this kind of role-based knowledge about processes from sentence-level descriptions in grade level texts. Semantic role labeling (SRL) systems can be trained to identify these process specific roles. However, these were developed for sentence-level interpretation and only ensure *within sentence* consistency of labels (Punyakanok et al., 2004; Toutanova et al., 2005; Lewis et al., 2015), limiting their ability to generate coherent characterizations of the process overall. In particular, the same process participant may appear in text at different syntactic positions, with different wording, and with different verbs, which makes it hard to extract globally consistent descriptions. In this work, we propose a cross sentence inference method to address this problem.

To illustrate the challenge consider some example sentences on *evaporation* shown in Table 1.The underlined spans correspond to fillers for an *undergoer* role i.e., the main entity that is undergoing evaporation. However, the filler *water* occurs as different syntactic arguments with different main actions. Without large amounts of process-specific training data, a supervised classifier will not able to

learn these variations reliably. Nevertheless, since all these sentences are describing *evaporation*, it is highly likely that *water* plays a single role. This expectation can be encoded as a factor during inference to promote consistency and improve accuracy, and is the basis of our approach.

We formalize this *cross sentence* joint inference idea as an Integer Linear Program (ILP). Our central idea is to collect all sentences for a single process, generate candidate arguments, and assign roles that are globally consistent for all arguments within the process. This requires a notion of consistency, which we model as pairwise alignment of arguments that should receive the same label. Argument-level entailment alone turns out to be ineffective for this purpose.

Therefore, we develop an alignment classifier that uses the compatibility of contexts in which the candidate arguments are embedded. We transform the original role-label training data to create alignment pairs from arguments that get assigned the same label, thus avoiding the need for additional labeling. Finally, the ILP combines the output of the SRL classifier and the alignment classifier in an objective function in order to find globally consistent assignments.

An empirical evaluation on a process dataset shows that proposed *cross sentence* formulation outperforms a strong *within sentence* joint inference baseline, which uses scores from a custom built role classifier that is better suited for the target domain.

In summary, this work makes the following contributions:

1. A cross-sentence, collective role-labeling and alignment method for harvesting process knowledge.

2. A high quality semantic resource that provides knowledge about scientific processes discussed in grade-level texts including physical, biological, and natural processes.

3. An evaluation which shows that the proposed cross sentence inference yields high quality process knowledge.

## 2 Related Work

Role-based representations have been shown to be useful for Open-domain factoid question answering (Shen and Lapata, 2007; Pizzato and Mollá, 2008), grade-level science exams (Jauhar et al., 2016) , and comprehension questions on process descriptions (Berant et al., 2014). Similar to process comprehension work, we target semantic representations about processes but we focus only on a high-level summary of the process, rather than detailed sequential representation of sub-events involved. Moreover, we seek to aggregate knowledge from multiple descriptions rather than understand a single discourse about each process.

There has been substantial prior work on semantic role labeling itself, that we leverage in this work. First, there are several systems trained on the PropBank dataset, e.g., EasySRL (Lewis et al., 2015), Mate (Björkelund et al., 2009), Generalized-Inference (Punyakanok et al., 2004). Although useful, the PropBank roles are verb (predicate) specific, and thus do not produce consistent labels for a *process* (that may be expressed using several different verbs). In constrast, frame-semantic parsers, e.g., SEMAFOR (Das et al., 2010), trained on FrameNet-annotated data (Baker et al., 1998) do produce concept (frame)-specific labels, but the FrameNet training data has poor ($< 50\%$) coverage of the grade science process terms. Building a resource like FrameNet for a list of scientific processes is expensive.

Several unsupervised, and semi-supervised approaches have been proposed to address these issues for PropBank style predicate-specific roles (Swier and Stevenson, 2004; Lang and Lapata, 2011; Fürstenau and Lapata, 2009; Fürstenau and Lapata, 2012; Lang and Lapata, 2010; Klementiev, 2012). A key idea here is to cluster syntactic signatures of the arguments and use the discovered clusters as roles. Another line of research has sought to perform joint training for syntactic parsing and semantic role labeling (Lewis et al., 2015), and in using PropBank role labels to improve FrameNet processing using pivot features (Kshirsagar et al., 2015).

Some SRL methods account for context information from multiple sentences (Ruppenhofer et al., 2010; Roth and Lapata, 2015). They focus on an-

| Process | Undergoer | Enabler | Action | Result |
|---|---|---|---|---|
| evaporation | liquid<br>water | heat<br>heat energy | changes<br>convert | gas<br>water vapor |
| weathering | rock<br>solid material | weather<br>heating | disintegration<br>breaking down | smaller rocks<br>smaller particles |
| photosynthesis | carbon dioxide<br>CO2 | solar energy<br>light energy | convert<br>transforms | energy<br>food |

**Table 2:** Examples of Target Knowledge Roles

notating individual event mentions in a document using discourse-level evidence such as co-reference chains. Our task is to aggregate knowledge about processes from multiple sentences in different documents. Although both tasks require raw SRL-style input, the different nature of the process task means that a different solution framework is needed.

Our goal is to acquire high quality semantic role based knowledge about processes. This allows us an unique opportunity to jointly interpret sentences that are discussing the same process. We build on ideas from previous within sentence joint inference (Punyakanok et al., 2004), argument similarity notions in semi and unsupervised approaches (Fürstenau and Lapata, 2012), and combining PropBank roles to propose a cross-sentence inference technique (Kshirsagar et al., 2015). The inference can be integrated with existing trained supervised learning pipelines, which can provide a score for role assignments for a given span.

## 3 Approach

Processes are complex events with many participating entities and inter-related sub-events. In this work, we aim for a relatively simple macro-level role-based knowledge about processes. Our task is to find classes of entities that are likely to fill key roles within a process namely, the *undergoer*, *enabler*, *result*, and *action*[1] (different verbs denoting the main action when the process is occurring, e.g., "dry"). We select these roles based on an initial analysis of grade science questions that involve recognizing instances of processes from their descriptions. Table 2 shows some examples of the target knowledge roles.

[1]For simplicity, we abuse the notion of a role to also include the main action as a role.

We develop a scalable pipeline for gathering such role-based process knowledge. The input to our system is the name of a process, e.g., "evaporate". Then we use a set of query patterns to find sentences that describe the process. A semantic role classifier then identifies the target roles in these sentences. The output is a list of typical fillers for the four process roles.

This setting presents a unique opportunity, where the goal is to perform semantic role labeling on a set of closely related sentences, sentences that describe the same process. This allows us to design a joint inference method that can promote expectations of consistency amongst the extracted role fillers.

There is no large scale training data that can be readily used for this task. Because we target process-specific and not verb-specific semantic roles, existing ProbBank (Kingsbury and Palmer, 2003) trained SRL systems cannot be used directly. Frame-semantic parsers trained on FrameNet data (Baker et al., 1998) are also not directly usable because FrameNet lacks coverage for many of the processes discussed in the science domain. Therefore, we create a process dataset that covers a relatively small number of processes, but demonstrate that the role extraction generalizes to previously unseen processes as well.

### 3.1 Cross-Sentence Inference

Given a set of sentences about a process, we want to extract role fillers that are globally consistent i.e., we want role assignments that are compatible. Our approach is based on two observations: First, any given role is likely to have similar fillers for a particular process. For instance, the undergoers of the evaporation process are likely to be similar – they are usually liquids. Second, similar arguments are

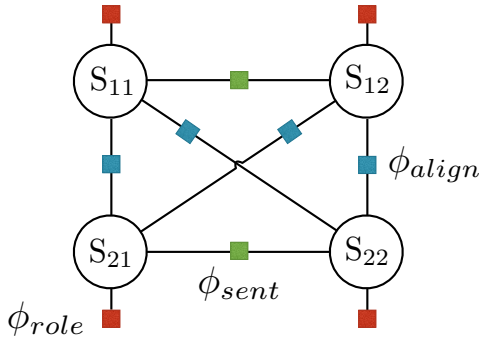


**Figure 1:** A factor graph representation of cross sentence inference. $S_{11}$ and $S_{12}$ denote role assignments for arguments $a_{11}$ and $a_{12}$ in one sentence, and $S_{21}$ and $S_{22}$ denote for arguments $a_{21}$ and $a_{22}$ in another. The $\phi_{role}$ factors score each role assignment to the arguments, and the $\phi_{align}$ factors score the compatibility of the connected arguments. $\phi_{sent}$ factors encode sentence level constraints.

unlikely to fill different roles for the same process. In evaporation, for example, it is highly unlikely that *water* is an undergoer in one sentence but is a result in another. These role-specific selectional preferences vary for each process and can be learned if there are enough example role fillers for each process during training (Zapirain et al., 2009; Zapirain et al., 2013). Since, we wish to handle processes for which we have no training data, we approximate this by modeling whether two arguments should receive the same role given their similarity and their context similarity.

Figure 1 illustrates the cross sentence inference problem using a factor graph. The $S_{ij}$ random variables denote the role label assignment for the $j^{th}$ argument in sentence $i$. Each assignment to an argument $S_{ij}$ is scored by a combination of the role classifier's score (factor $\phi_{role}$), and its pairwise compatibility with the assignments to other arguments (factor $\phi_{align}$). The factors $\phi_{sent}$ capture two basic within sentence constraints.

### 3.2 Inference using ILP

We formulate the cross sentence inference task using an Integer Linear Program shown in Figure 2. The ILP seeks an assignment that maximizes a combination of individual role assignment scores and their global compatibility, which is measured as the similarity of fillers for the same role minus similarity of

$$\arg\max_{\mathbf{z}} \sum_k \sum_{i,j} z_{ijk} \Bigg( \lambda \underbrace{\phi_{role}(a_{ij}, k)}_{\text{Role classifier score}} + (1-\lambda) \underbrace{\Big[ \Delta(a_{ij}, k) - \nabla(a_{ij}, k) \Big]}_{\text{Global compatibility}} \Bigg)$$

where compatibility with same roles is:

$$\Delta(a_{ij}, k) = \frac{1}{\tilde{N}_k} \sum_{l,m} z_{lmk} \phi_{align}(a_{ij}, a_{lm})$$

and compatibility with other roles is:

$$\nabla(a_{ij}) = \frac{2}{\tilde{N}_{k'}} \sum_{l,m} \sum_{n \neq k} z_{lmn} \underbrace{\phi_{align}(a_{ij}, a_{lm})}_{\text{Penalty when role n} \neq \text{k}}$$

subject to:

$$\sum_k z_{ijk} \leq 1 \qquad \forall\ a_{ij} \in \text{sentence}_i$$

$$\sum_j z_{ijk} \leq 1 \qquad \forall\ a_{ij} \in \text{sentence}_i, k \in \text{R}$$

$\tilde{N}_k$ : Approximate number of arguments with role $k$

$\tilde{N}_{k'}$ : Approximate number of arguments with role $n \neq k$

**Figure 2:** An Integer Linear Program formulation of the Cross-sentence Inference.

fillers of different roles.

The decision variables $z_{ijk}$ denote role assignments to arguments. When $z_{ijk}$ is set it denotes that argument $j$ in sentence $i$ ($a_{ij}$) has been assigned role $k$. The objective function uses three components to assign scores to an assignment.

1. Classifier Score $\phi_{role}(a_{ij}, k)$ – This is the score of a sentence-level role classifier for assigning role $k$ to argument $a_{ij}$.
2. Within Role Compatibility $\Delta(a_{ij}, k)$ – This is a measure of argument $a_{ij}$'s compatibility with other arguments which have also been assigned the same role $k$. We measure compatibility using a notion of alignment. An argument is said to align with another if they are similar to each other in some respect (either lexically or semantically). As we show later, we develop an alignment cclassifier which predicts an alignment score $\phi_{align}$ for each pair of arguments. The compatibility is defined as a normalized sum of the alignment scores for argument $a_{ij}$ paired with

other arguments that have also been assigned the role $k$. Without some normalization roles with many arguments will receive higher compatibility scores.To avoid this, we normalize by $(1/\tilde{N}_k)$, where $\tilde{N}_k$ refers to the number of arguments that the role classifier originally labeled with role $k$, an approximation to the number of arguments that are currently assigned role $k$ by the ILP.

3. Across Role Incompatibility $\nabla(a_{ij}, k)$ – This is a measure of how well $a_{ij}$ aligns with the other arguments that are assigned a different role ($n \neq k$). For good assignments this quantity should be low. Therefore we add this as a penalty to the objective. As with $\Delta$, we use an approximation for normalization $(1/\tilde{N}_{k'})$, which is the product of other roles and the number of arguments in other sentences that can receive these roles. Because $\tilde{N}_{k'}$ is typically higher, we boost this score by 2 to balance against $\Delta$.

Last, we use two sets of hard constraints to enforce the standard within-sentence expectations for roles: 1) A single argument can receive only one role label, and 2) A sentence cannot have more than one argument with the same label, except for the NONE role.

We use an off-the-shelf solver in Gurobi (www.gurobi.com) to find an approximate solution to the resulting optimization problem.

### 3.3 Role Classifier ($\Phi_{role}$)

The role classifier provides a score for each role label for a given argument. Although existing SRL and frame semantic parsers do not directly produce the role information we need (Section 2), we build on them by using their outputs for building a process role classifier.

Before we can assign role labels, we first need to generate candidate arguments. Using EasySRL (Lewis et al., 2015), a state-of-the-art SRL system, we generate the candidate argument spans for each predicate (verbs) in the sentence. Then, using a linear SVM classifier (Fan et al., 2008), we score the candidate arguments and the predicates for our four roles and a special NONE role to indicate the argument is not one of the four. The classifier is trained with a set of annotated examples (see Section 4) with the following sets of features.

i) Lexical and Syntactic – We use a small set of standard SRL features such as lexical and syntactic contexts of arguments (e.g., head word, its POS tag) and predicate-argument path features (e.g, dependency paths). We also add features that are specific to the nature of the process sentences. In particular, we encode syntactic relationships of arguments with respect to the process name mention in the sentence. We use Stanford CoreNLP toolkit (Manning et al., 2014) to obtain POS tags, and dependency parses to build these features.

ii) PropBank roles – While they do not have a 1-to-1 correspondence with process roles, we use the EasySRL roles coupled with the specific predicate as a feature to provide useful evidence towards the process role.

iii) Framenet Frames – We use the frames evoked by the words in the sentence to allow better feature sharing among related processes. For instance, the contexts of undergoers in evaporation and condensation are likely to be similar as they are both state changes which evoke the same `Undergo_Change` frame in FrameNet.

iv) Query patterns – We use query patterns to find sentences that are likely to contain the target roles of interest. The query pattern that retrieved a sentence can help bias the classifier towards roles that are likely to be expressed in it.

### 3.4 Alignment Classifier ($\Phi_{align}$)

Our goal with alignment is to identify arguments that should receive similar role labels. One way to do this argument alignment is to use techniques developed for phrase level entailment or similarity which often use resources such as WordNet and distributional embeddings such as word2vec (Mikolov et al., 2013) vectors. It turns out that this simple entailment or argument similarity, by itself, is not enough in many cases for our task[2]. Moreover, the enabler, and the result roles are often long phrases whose text-based similarity is not reliable. A more robust approach is necessary. Lexical and syntactic similarity of arguments and the context in which they are embedded can provide valuable additional information. Table 3 shows a complete list of features we use to train the alignment classifier.

[2]We used an approach that combined WordNet-based phrase similarity method, and Word2Vec vector similarity, where the vectors were learned from a general news domain.

| **Lexical** |
|---|
| Entailment score of arguments.<br>Word2vec similarity of argument vectors.<br>Word2Vec similarity of head nodes of arguments.<br>Word2Vec similarity of parent of the head nodes.<br>Word2Vec similarity of verbs of argument sentences.<br>Jaccard similarity of children of the head node. |
| **Syntactic** |
| Similarities of frames to right and left of arguments.<br>Jaccard similarity of POS tags of argument.<br>POS tag of head nodes match (boolean).<br>POS tag of head node parents match (boolean).<br>Similarity of dep. path from arg to process name.<br>Similarity of POS tags on arg to process name path.<br>Similarity of POS tags of arg's children.<br>Similarity of the dependencies of the arg's head. |
| **Sentence** |
| Query patterns match argument sentences (boolean). |

**Table 3:** Alignment Classifier Features. Similarities of sets were calculated using Jaccard co-efficient.

Fortunately, learning this classifier does not require any additional training data. The original data with annotated semantic role labels can be easily transformed to generate supervision for this classifier. For any given process, we consider all pairs of arguments in different sentences (i.e., $(a_{ij}, a_{lm})$ : $i \neq l$) and label them as aligned if they are labeled with the same role, or unaligned otherwise.

## 4 Evaluation

Our goal is to generate knowledge about processes discussed in grade-level science exams. Since existing semantic resources such as FrameNet do not provide adequate coverage for these, we created a dataset of process sentences annotated with the four process roles: undergoer, enabler, action, and result.

### 4.1 Dataset

This dataset consists of 1205 role fillers extracted from 537 sentences retrieved from the web. We first compiled the target processes from a list of process-oriented questions found in two collections: (i) New York Regents science exams (Clark, 2015), and (ii) helpteaching.com, a Web-based collection of practice questions. Then, we identified 127 process questions from which we obtained a set of 180 unique target processes. For each target process, we queried the web using Google to find definition-style sentences, which describe the target process. For each process we discarded some noisy sentences through a combination of automatic and manual filtering.

| Query Patterns |
|---|
| ⟨name⟩ is the process of ⟨x⟩ |
| ⟨name⟩ is the process by which ⟨x⟩ |
| ⟨name⟩ {occurs when} ⟨x⟩ |
| ⟨name⟩ { helps to \| causes } ⟨x⟩ |

**Table 4:** Example query patterns used to find process description sentences.

Table 4 shows some examples of the 14 query patterns that we used to find process descriptions. Because these patterns are not process-specific, they work for unseen processes as well.

To find role fillers from these sentences, we first processed each sentence using EasySRL (Lewis et al., 2015) to generate candidate arguments. Some of the query patterns can be used to generate additional arguments. For example, in the pattern "⟨name⟩ is the process of ⟨x⟩" if ⟨x⟩ is a noun then it is likely to be an undergoer, and thus can be a good candidate. [3]. Then two annotators annotated the candidate arguments with the target roles if one were applicable and marked them as NONE otherwise. Disagreements were resolved by a third annotator. The annotations spanned a random subset of 54 target processes. The role label distribution is shown below:

| Role | No. of instance |
|---|---|
| Undergoer | 77 |
| Enabler | 154 |
| Action | 315 |
| Result | 194 |
| NONE | 465 |

**Table 5:** Role distribution

We conducted five fold cross validation experiments to test role extraction. To ensure that we are testing the generalization of the approach to unseen

[3]These patterns are ambiguous and are not adequate by themselves for accurately extracting the roles. We use them as features.

processes, we generated the folds such that the processes in the test fold were unseen during training. We compared the basic role classifier described in Section 3.3, the *within sentence* and the *cross sentence* inference models. We tune the ILP parameter $\lambda$ for cross sentence inference based on a coarse-grained sweep on the training folds.

We also compared with a simple baseline that learned a mapping from PropBank roles produced by EasySRL system to the process roles by using the roles and the verb as features. We also add the FrameNet frames invoked by the lexical unit in the sentence. Note this is essentially a subset of the features we use in our role classifier. As a second baseline, we compare with a (nearly) out-of-the-box application of SEMAFOR (Das et al., 2010), a FrameNet based frame-semantic parser. We modified SEMAFOR to override the frame identification step since the process frame information is already associated with the test sentences.

### 4.2 Cross-Sentence Inference Results

Table 6 compares performance of the different methods. The learned role mapping of shallow semantic roles performs better than SEMAFOR but worse than the simple role classifier. SEMAFOR uses a large set of features which help it scale for a diverse set of frames in FrameNet. However, many of these many not be well suited for the process sentences in our relatively smaller dataset. Therefore, we use our custom role classifier as a strong baseline to demonstrate within and cross sentence gains. Enforcing sentence-level consistency through joint inference, shown as (+within sent.), improves over the baseline which does not use any consistency. It gains precision (by nearly 8 points), while loosing recall in the trade-off (by about 4.7 points) to yield an overall gain in F1 by 1.05 points. Enforcing cross sentence consistency, shown as (+cross sent.) provides additional gains beyond within sentence inference by another 2.38 points in F1 [4]. Table 7 shows how the gains are distributed across different roles. Cross sentence inference provides improvements for all roles, with the biggest for undergoers (nearly 4 points).

| Method | Prec. | Rec. | F1 |
|---|---|---|---|
| Role mapping | 56.62 | 59.60 | 58.07 |
| SEMAFOR | 40.72 | 50.54 | 45.10 |
| Role class. ($\phi_{role}$) | 78.48 | **78.62** | 78.55 |
| + within sent. | 86.25 | 73.91 | 79.60 |
| + cross sent. | **89.84** | 75.36 | **81.97**†† |

**Table 6:** Process role inference performance. †† indicates significant improvement over Role + within sentence system.

| Method | Und. | Ena. | Act. | Res. |
|---|---|---|---|---|
| Role Class. | 65.38 | 73.84 | 83.58 | 77.30 |
| + within | 66.01 | 73.11 | 86.70 | 76.11 |
| + cross | **70.00** | **74.31** | **89.30** | **78.00** |

**Table 7:** Performance (F1) across all roles.

Figure 3 shows the precision/recall plots for the basic role classifier and within and cross sentence inference. Both inference models trade recall for gains in precision. Cross sentence yields higher precision at most recall levels, for a smaller overall loss in recall compared to within sentence (1.6 versus 4.9).

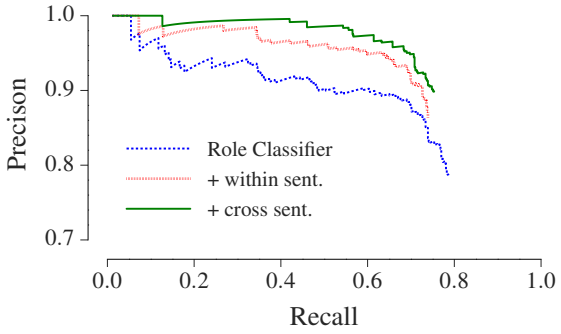


**Figure 3:** Precision/Recall trade-offs for process role inference. y-axis is truncated at 0.7 to better visualize the differences.

### 4.3 Ablations

Table 8 shows the utility of various components of cross sentence inference. Using argument entailment alone turns out to be ineffective and only produces a minor improvement (0.16 in F1). However, the alignment classifier scores are much more effective and yield about 2.37 points gain in F1. Both within and across role compatibilities, $\Delta$ and $\nabla$, yield statistically significant improvements[5] over

[4] The single parameter in ILP turned out to be stable across the folds and obtained this best value at $\lambda = 0.8$.

[5] Significance measured using approximate randomization test

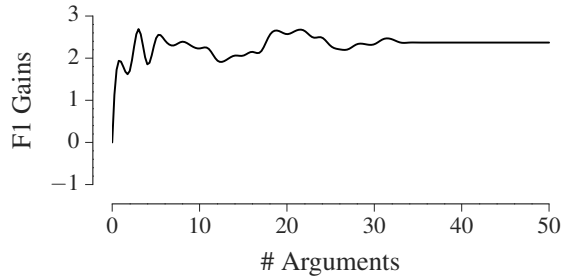


**Figure 4:** Cross sentence gains in F1 when varying the number of most similar arguments used to assess compatibilities.

within sentence inference. Combining these complementary compatibilities provides the best gains.

| Method | Prec. | Rec. | F1 |
|---|---|---|---|
| within sent. | 86.25 | 73.91 | 79.60 |
| **+ Entailment** | | | |
| cross sent. w/ $\Delta$ | 85.13 | 72.64 | 78.39 |
| cross sent. w/ $\nabla$ | 85.98 | 73.36 | 79.17 |
| cross sent. w/ $\Delta + \nabla$ | 86.62 | 73.91 | 79.76 |
| **+ Alignment Classifier** | | | |
| cross sent. w/ $\Delta$ | 89.07 | 75.36 | 81.64†† |
| cross sent. w/ $\nabla$ | 88.72 | 75.54 | 81.60†† |
| cross sent. w/ $\Delta + \nabla$ | **89.84** | 75.36 | **81.97**†† |

**Table 8:** Performance impact of inference components. †† indicates significant improvement over within sentence.

We also studied the effect of varying the number of arguments that ILP uses to measure the compatibility of role assignments. Specifically, we allow inference to use just the top k alignments from the alignment classifier. Figure 4 shows the main trend. Using just the top similar argument already yields a 1 point gain in F1. Using more arguments tends to increase gains in general but with some fluctuations. Finding an assignment that respects all compatibilities across many argument pairs can be difficult. As seen in the figure, at some of the shorter span lengths we see a slightly larger gain ($+0.3$) compared to using all spans. This hints at benefits of a more flexible formulation that makes joint decisions on alignment and role label assignments.

Table 9 shows an ablation of the alignment classifier features. Entailment of arguments is the most informative feature for argument alignment. Adding lexical and syntactic context compatibilities adds significant boosts in precision and recall. Knowing that the arguments are retrieved by the same query pattern (sentence feature) only provides minor improvements. Even though the overall classification performance is far from perfect, cross sentence can benefit from alignment as long as it provides a higher score for argument pairs that should align compared to those that should not.

| Feature | P | R | F1 |
|---|---|---|---|
| Entailment score only | 39.55 | 14.59 | 21.32 |
| +Lexical | 50.75 | 26.02 | 34.40 |
| +Syntactic | 62.31 | 31.47 | 41.82 |
| +Sentence | 62.33 | 31.41 | 41.53 |

**Table 9:** Performance of different feature groups for alignment.

#### 4.3.1 Error Analysis

We conduct an error analysis over a random set of 50 errors observed for cross sentence inference. In addition to issues from noisy web sentences and nested arguments from bad candidate extraction, we find the following main types of errors:

- Dissimilar role fillers (27.5 %) – In some processes, the fillers for the *result* role have high levels of variability that makes alignment error prone. For the process camouflage, for instance, the *result* roles include 'disorientation', 'protect from predator', 'remain undetected' etc.
- Bad role classifier scores (37.5%) – For some instances the role classifier assign high scores to incorrect labels, effectively preventing the ILP from flipping to the correct role. For example, the argument that follows "causes" tends to be a result in many cases but not always, leading to high scoring errors. For example, in the sentence with "...when heat from the sun causes water on earth's ...", the role classifier incorrectly assigns 'water' to a result role with high confidence.
- Improper Weighting (7.5%)– Sometimes the ILP does not improve upon a bad top choice from the role classifier. In some of these cases, rather than the fixed lambda, a different weighted combination of role and alignment classifier scores

would have helped the ILP to flip. For example, the argument 'under autumn conditions' from the sentence 'hibernation occurs when the insects are maintained under autumn conditions.' has a good role score and is similar to other correctly labeled enablers such as 'cold , winter conditions' but yet is unable to improve.

The rest (27.5 %) are due to noisy web sentences, incorrect argument extraction and errors outside the scope of cross sentence inference.

## 5 Conclusions

Simple role-based knowledge is essential for recognizing and reasoning about situations involving processes. In this work we developed a cross sentence inference method for automatically acquiring such role-based knowledge for new processes. The main idea is to enforce compatibility among roles extracted from sentences belonging to a single process. We find that the compatibility can be effectively assessed using an alignment classifier built without any additional supervision. Empirical evaluation on a process dataset shows that cross sentence inference using an Integer Linear Program helps improve the accuracy of process knowledge extraction.

## 6 Acknowledgement

The authors would like to thank the anonymous reviewers for helpful comments, Meghana Kshirsagar, Sam Thomson, Mike Lewis for answering implementation details of their systems, and the Stony Brook NLP Lab members for their valuable feedback and suggestions. This work is supported in part by Foreign Fulbright PhD Fellowship and by the grant from Allen Institute for Artificial Intelligence.

## References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Abby Vander Linden, Brittany Harding, Brad Huang, Peter Clark, and Christopher D. Manning. 2014. Modeling biological processes for reading comprehension. In *Proceedings of EMNLP*.

Anders Björkelund, Love Hafdell, and Pierre Nugues. 2009. Multilingual semantic role labeling. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 43–48. Association for Computational Linguistics.

Peter Clark. 2015. Elementary school science and math tests as a driver for ai: Take the aristo challenge. *to appear*.

Dipanjan Das, Nathan Schneider, Desai Chen, and Noah A. Smith. 2010. Probabilistic frame-semantic parsing. In *Proc. of NAACL-HLT*.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Hagen Fürstenau and Mirella Lapata. 2009. Graph alignment for semi-supervised semantic role labeling. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 11–20, Singapore.

Hagen Fürstenau and Mirella Lapata. 2012. Semi-supervised semantic role labeling via structural alignment. *Computational Linguistics*, 38(1):135–171.

Sujay Kumar Jauhar, Peter D. Turney, and Eduard H. Hovy. 2016. Tables as semi-structured knowledge for question answering. In *ACL*.

Paul Kingsbury and Martha Palmer. 2003. Propbank: the next level of treebank. In *Proceedings of Treebanks and lexical Theories*, volume 3. Citeseer.

Ivan Titov Alexandre Klementiev. 2012. Semi-supervised semantic role labeling: Approaching from an unsupervised perspective. In *Proceedings of the COLING Conference.*

Meghana Kshirsagar, Sam Thomson, Nathan Schneider, Jaime G. Carbonell, Noah A. Smith, and Chris Dyer. 2015. Frame-semantic role labeling with heterogeneous annotations. In *ACL*.

Joel Lang and Mirella Lapata. 2010. Unsupervised induction of semantic roles. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 939–947, Los Angeles, California, June. Association for Computational Linguistics.

Joel Lang and Mirella Lapata. 2011. Unsupervised semantic role induction with graph partitioning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1320–1331, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Mike Lewis, Luheng He, and Luke Zettlemoyer. 2015. Joint a* ccg parsing and semantic role labelling. In *Empirical Methods in Natural Language Processing*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit.

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

Luiz Augusto Pizzato and Diego Mollá. 2008. Indexing on semantic roles for question answering. In *Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering*, pages 74–81. Association for Computational Linguistics.

Vasin Punyakanok, Dan Roth, Wen-tau Yih, and Dav Zimak. 2004. Semantic role labeling via integer linear programming inference. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Michael Roth and Mirella Lapata. 2015. Context-aware frame-semantic role labeling. *Transactions of the Association for Computational Linguistics (TACL)*, 3:449–460.

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 Evaluation*, SemEval '10, pages 45–50, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dan Shen and Mirella Lapata. 2007. Using semantic roles to improve question answering. In *EMNLP-CoNLL*, pages 12–21.

Robert S Swier and Suzanne Stevenson. 2004. Unsupervised semantic role labelling. In *EMNLP*.

Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. 2005. Joint learning improves semantic role labeling. In *ACL*.

Beñat Zapirain, Eneko Agirre, and Lluís Màrquez i Villodre. 2009. Generalizing over lexical features: Selectional preferences for semantic role classification. In *ACL*.

Beñat Zapirain, Eneko Agirre, Lluís Màrquez i Villodre, and Mihai Surdeanu. 2013. Selectional preferences for semantic role classification. *Computational Linguistics*, 39:631–663.