Semantic Role Labeling for Process Recognition Questions

Samuel Louvan[+], Chetan Naik[+], Veronica Lynn[+], Ankit Arun[+], Niranjan Balasubramanian[+], Peter Clark[*]

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

Abstract

We consider a 4th grade level question answering task. We focus on a subset involving recognizing instances of physical, biological, and other natural processes. Many processes involve similar entities and are hard to distinguish using simple bag-of-words representations alone.

Simple semantic roles such as *Input*, *Result*, and *Enabler* can often capture the most critical bits of information about processes. Our QA system scores answers by aligning semantic roles in the question against the roles in the knowledge. Empirical evaluation shows that manually generated roles provide a 12% relative improvement in accuracy over a simpler bag-of-words representation. However, automatic role identification is noisy and doesn't provide gains even with distant supervision and domain adaptation modifications to account for the limited training data. In addition, we conducted an error analysis of the QA system when using the manual roles. We find representational gaps i.e., cases where critical information doesn't fit into any of the current roles, as well as entailment issues that motivate deeper reasoning beyond simple role based alignment for future work.

1 Introduction

Grade-level science exams are an excellent benchmark for developing knowledge-driven question answering systems (Clark et al., 2013). These exams test students' understanding of a wide variety of concepts including physical, biological, and other natural processes. Some questions test the ability to recognize a process given a description of its instance. Here is an example: *A puddle is drying in the sun. This is an example of A) evaporation, B) condensation, C) melting, D) sublimation.* This work explores a knowledge-driven approach to answering such questions.

In particular, we investigate a light-weight semantic role based representation to answer 4th grade science questions that test process recognition. Many processes are similar to each other and not surprisingly are often described using similar words. For instance, both *evaporation* and *condensation* are both phase changes involving liquids and gases. Distinguishing between these two requires an understanding of the different roles that liquids and gases play in these processes. This type of knowledge is naturally expressed via semantic roles.

We design a light-weight representation that applies to many processes. In general a process has an input – the artifact that undergoes the process, an output – the artifact that results from the process, and an enabler – an artifact or event that helps the process. In addition we also include key actions that denote the main events in the process. For example, in evaporation the typical input is a liquid and the output is a gaseous form of the input substance. The enabler is some form of heat source such as the sun. [1]

Given this semantic role based representation, recognizing a process instance (i.e., answering a question) becomes a task of assessing how well the roles of the instance in the question align with the typical roles of the candidate answer processes.

Our preliminary experiments show that manually generated semantic representations can provide more than a 12% relative improvement in QA performance over a simple bag-of-words representation. A scalable solution however requires automatic extraction. We investigated an off-the-shelf semantic role labeling system, MATE (Björkelund et al., 2009), to extract these rep-

[1]This limited representation is incomplete and semantically inaccurate in some cases. For instance, we ignore sub-events and any sequence or order information between them. We chose this representation as it covers a majority of the questions at the 4th grade level and is also more amenable to automatic extraction.

resentations and evaluated their performance in QA.

SRL systems such as MATE are supervised systems that require large amounts of training data. Unfortunately, existing resources such as FrameNet do not cover all our target scientific processes. We manually created a small amount of training data using sentences that involve processes mentioned in the questions. To account for the limited amount of training data, we explored distant supervision, and domain adaptation. We find that domain adaptation yields a modest improvement, while distant supervision is unreliable. Overall, we find that automatic extraction is still quite noisy and as a result doesn't provide improvements over using simple bag-of-words representations.

An error analysis shows the knowledge gaps and the deficiencies in the current representation and the key linguistic phenomenon that lead to errors in automatic SRL, which present interesting avenues for future work.

2 Representing Processes via Semantic Roles

A portion of the grade science exams test the ability to recognize the instances of physical, biological, and other natural processes. The questions present a short description of an instance and multiple process names as the answer choices.

Table below shows a few examples:

1) As water vapor rises in the atmosphere, it cools and changes back to liquid. Tiny drops of liquid form clouds in this process called (A) condensation (B) evaporation (C) precipitation (D) run-off.
2) The process of change from an egg to an adult butterfly is an example of: (A) vibrations (B) metamorphosis.
3) A new aluminum can made from used cans is an example of: (A) recycling (B) reducing (C) reusing (D) repairing

The descriptions of the instances are often short. They mention the main entities involved and the change brought about by the process. At the 4th grade level, the questions do not involve deep knowledge about the sub-events or their sequential order. Rather the questions test for shallower knowledge about the entities undergoing change, the resulting artifacts, and the main characteristic action describing the process. This knowledge is naturally expressed via semantic roles. Accordingly we design a simple representation that encodes information about each process via the following roles:

1. *Input* – This role captures the main input to the process or the object undergoing the process. e.g., Water is an input to the evaporation process.
2. *Result* – The artifact that results from the process or the change that results from the process e.g., Water vapor is a result of evaporation.
3. *Trigger* – The main action, expressed as a verb or its nominalization, indicating the occurrence of the process. e.g., *converted* is a trigger for evaporation.
4. *Enabler* – The artifact, condition or action that enables the process to happen. e.g., *Sun* is a heat source that is enabler for evaporation.

Our goal is to build aggregate knowledge about processes from multiple sentences. In effect we wish to create a table, whose columns are the semantic roles and rows are role fillers obtained from sentences mentioning the process. We gather knowledge about processes from both definitional sentences and from sentences that describe instances of the processes.

Definitional sentences present *type* information about the roles. For instance, the input to evaporation is typically a liquid. During question answering, we have to check type compatibility of the roles in the question and the definitional sentence. Having instance level information in the process knowledge can help situations where type resolution fails. On the other hand, having definitional sentences provides better coverage of information about roles compared to instance sentences, which can sometimes omit roles that are obvious.

2.1 MATE System for SRL

There have been various SRL approaches , in terms of resources (PropBank, FrameNet) and also the techniques and features used. Nevertheless, in general the workflow of an SRL consist of two parts namely argument identification and argument classification. In this work, we use MATE SRL system (Björkelund et al., 2009). The reason we use MATE in our setting is because the code is publicly available, it provides automatic predicate identification, and it is one of the high performing system in ConLL'09 SRL shared task for English dataset.

MATE accepts sentences in CoNLL'09 format and processes them through several pipelines

namely predicate identification, argument identification, and argument classification. The features used in the pipeline are based on the syntactic information obtained from POS tagger and dependency parser such as the position of the argument with respect to the predicate, the dependency path between the arguments and the predicate, the set of POS tag of the predicate's children etc. Please refer to (Björkelund et al., 2009) for details.

The modification that we make to MATE are related to predicate identification and domain adaptation. For the predicate identification, in case of the classifier fails to identify one, we force the classifier to output one predicate based on the sorted confidence score of each of the predicate candidates. For the domain adaptation part, we adopt the simple feature augmentation approach (Daumé III, 2009).

Structured prediction problems such as SRL require substantial amounts of training data. SRL systems such as MATE typically train on resources such as FrameNet, which contain more than 190,000 sentences. Semantic roles are not reliably identified with syntactic patterns alone. A pattern such as *[verb] to [X]* could suggest that X is a result or enabler depending on the process at hand. For instance, *change to [X]* indicates a resulting state, whereas *adapts to [X]* doesn't. This suggests that lexical information (e.g., the type of verb) is quite critical. Unless the lexical variations had all been observed in the training data, generalization is likely to suffer. We explore two ideas to address this issue.

2.2 Domain Adaptation

A straightforward approach to training the SRL system is to combine all process sentences into one pool and learn a single model. As an alternate approach, we can learn a SRL model for every process using only the sentences that describe the process. This enables learning from a much smaller but highly relevant set of sentences. Note this is possible in our setting because we know beforehand which process each sentence is describing. Rather than picking one approach over the other, we can combine their strengths using domain adaptation ideas (Daumé III, 2009).

The sentences that describe the target process can be viewed as target domain data, and the sentences describing all other processes can be viewed as the source domain data. Following (Daumé III, 2009) we utilize a simple approach for domain adaptation. The key idea is to take the existing features and create a new feature vector that contains three versions of the original features. A source-specific version, a target-specific version, and a general version. The instances from the source domain will only contain the general and source-specific versions, and the instances for the target domain will contain the general and the target-specific versions. Formally, if $\mathbf{f}$ is the set of features used, then we use the following mappings to create new feature vectors:

$$\Phi^s(f) =< f, f, 0 >$$
$$\Phi^t(f) =< f, 0, f >$$

where, Φ^s is applied to source sentences and Φ^t is applied to target sentences. This mapping enables the learning algorithm to do domain adaptation by learning two sets of weights that reflect the utility of a feature across all domains as well as within the target domain. This simple transformation has been shown to be quite effective for domain adaptation (Daumé III, 2009).

2.3 Distant Supervision

Distant supervision (Mintz et al., 2009), is an increasingly popular method for relation extraction and typically used in a setting where we have a lot of unlabeled data and there exist the source of labeled knowledge base. They key assumption of distant supervision is " if two entities participate in a relation, any sentence that contain those two entities might express that relation" (Mintz et al., 2009).

We make a similar assumption and use distant supervision to obtain more labeled data. We use a the Waterloo corpus [2] as our unlabeled data source. We search this corpus to find sentences related to each process and automatically annotate them for semantic roles. We use the following procedure:

- Collect most frequent role fillers for each processes. Create a query that finds sentences containing the role fillers.
- For each retrieved sentence, identify the location of each role filler. Label a candidate role filler only if there is a up-path $\uparrow$ (dependency path) from the candidate role filler nodes to the trigger node. If there is no such path then skip the sentence.
- We also use simple hand crafted lexical patterns to identify *Input, Enabler and Result.*

[2] This is a large collection of Web documents about 280 GB collected by Charlie Clark at Univ. of Waterloo.

For example, *Enablers* are often characterized with the preceding words such as *by, through,with, because*. *Results* usually appear after the word *causes, into, produce*.

3 Question Answering Using Semantic Roles

In this section, we describe our approach to answering process recognition questions using the semantic role based representation of processes. The questions describe a specific instance of a process or a prototypical occurrence of a process. The QA task is then to choose the correct process that is being described. Intuitively, we can score each process based on how well the roles of the instance described in the question align with the roles of the process we've gathered from other sentences describing the process. Higher alignment suggests better evidence. In particular, use the following procedure to answer questions:

1. Convert the question into k statements $(A_1, ..., A_k)$, one for each answer option.
2. Identify the roles in each question statement by processing them through MATE.
3. For each statement A_i, collect the corresponding process rows from the process knowledge table $(S_1, ..., S_l)$.
4. For each row in the table compute an alignment score by checking for the textual entailment of the corresponding roles.

$$alignment(A, S) = \sum_{role_i \in R} entails(role_i(A), role_i(S))$$

   where, R ={Input, Result, Enabler, Trigger}. $entails(x, y)$ is computed as a textual entailment score that reflects how well the text x entails text y or the other way around. It measures the coverage of the statement role accounting for synonyms and hypernyms using WordNet.
5. Compute the mean of the top 5 alignment scores and use it as the final score for each process.
6. Return the top scoring process as the answer.

4 Evaluation

4.1 Process Recognition Collection

Our data set contains 141 process identification questions, which were selected manually out of 4650 questions from Help Teaching[3], a collection of tests and worksheets for parents and educators, and 195 questions from the 4th-grade New York Regents Science Exams collection, similar to the one used in (Clark et al., 2013). We selected multiple-choice questions where at least two of the answer choices, including the correct answer, were processes.

For each question, we identified all processes given as answer choices and collected definition sentences for each. In total, we collected 948 definition sentences covering 183 processes. These sentences were obtained from a variety of sources such as Barron's Study Guides and various web resources including Wikipedia and WordNet.

Each question and definition sentence was annotated by following a set of guidelines to identify the roles expressed by the sentence. A sentence could contain multiple (or no) values for a single role, but text spans are not allowed to overlap - that is, the same word or phrase cannot be used for multiple roles. Disagreements were resolved by a second annotator.

4.2 Semantic Role Labeling

First, we compare the performance of SRL using MATE under the following configurations:

- *Standard* – This configuration uses sentences from all processes combined together for training and does not distinguish at test time whether the sentence is describing a particular process.
- *Per-Process* – This uses only the target process sentences for training. This setting requires that we have some seed set of sentences for every process.
- *Domain Adaptation* – This uses both target process sentences as well other sentences using the domain adaptation technique described in Section 2.2.
- *Distant Supervision* – This setting uses the sentences obtained via distant supervision for training. However, it trains only on the sentences that describe the target process and does not use sentences describing the other processes. This requires that we have some seed knowledge for each process but not necessarily sentence-level annotations.

We use 5-fold cross validation to test each configuration. Table 1 shows the results. The results show that MATE is not highly effective with the best F1 of around 0.38. This is sub-

[3]http://www.helpteaching.com

Method	Precision	Recall	F1
Standard	0.4323	0.3325	0.3758
Per Process	0.4225	0.2556	0.3185
Distant Supervision	**0.5614**	0.2642	0.3594
Dom. Adaptation	0.4386	**0.3351**	**0.3799**

Table 1: Semantic Role Labeling Performance. Bold face entries indicate the best performance.

stantially lower compared to the state-of-the-art performance of MATE on standard datasets such as CoNLL, where the performance around 0.80 in F1. We believe that the limited amount of training data is a key factor in the performance differences. Training on target domain sentences using the Per-Process model results in lower performance, whereas Standard, Distant Supervision, and Domain Adaptation, all with increased amounts of data result in better performance.

We find that Domain Adaptation provides minor improvements over the Standard model. Interestingly, Distant Supervision doesn't improve F1 but achieves the best overall precision. We hypothesize that this is in part due to the change in distribution of the roles. Many Distant Supervision annotated sentences do not contain all the roles. Thus, the prior probability of observing any particular role decreases potentially causing the learner to be more conservative about predicting roles.

Method	Accuracy
BOW	63.12
Manual SRL	67.38
BOW+Manual SRL	**70.92**
Standard	**55.32**
Per Process	46.80
Domain Adaptation	**55.32**
Distant Supervision	51.77
BOW + Standard	**65.24**

Table 2: Question Answering Performance. Bold face indicates the best performance in each block.

4.3 QA

Our central hypothesis is that using semantic-role based representation helps in process recognition questions. We compare manually generated semantic roles (as a result of the annotation process), and the performance of automatically derived semantic roles for question answering. We use the process described in Section 3 for answering questions.

Table 2 shows the QA accuracy when using semantic roles generated by the different configurations. As a baseline, we use a simple bag-of-words system that uses textual entailment but without the semantic roles (BOW). Manual SRL, is a system which uses manually assigned semantic roles as its representation. We find that using manual roles by itself for alignment yields a 9% relative improvement in accuracy. BOW and SRL perform well in slightly different subsets of questions. As a result combining SRL-based scoring with BOW based scoring yields more than 12% improvement. While access to semantic roles provides gains, a variety of other issues also need to be addressed to further improve QA performance (Section 4.4).

Automatically obtained semantic roles using the Standard formulation is worse than BOW by itself. However, in conjunction with BOW, Standard SRL provides minor gains over using BOW alone. For the most part, the QA performance of the other automatic SRL variants track the SRL extraction accuracy. This suggests that improving the automatic SRL performance is likely to provide improvements in QA.

4.4 Error Analysis

Automatic SRL Failures

A qualitative analysis of the SRL errors showed two main issues that arise out of data sparsity. Much of the errors arise from failures to identify the proper predicate. It turns out that the dominant pattern is where the predicate is a verb that is directly attached to the ROOT. Other syntactic patterns are infrequently observed and are not very reliable. Automatic SRL fails in these cases. Similarly the dependency labels of the children of a node are also strong features for predicate prediction but this feature again fails except in cases of the most dominant dependencies.

A closer investigation suggests much of the variation in these patterns arise because of functional phrases such as "is a process by which". Simplifying the sentences by eliminating these constructions can lead to improved performance.

QA Failures

Even with manually assigned roles, the QA sys-

tem failed on nearly 30% of the questions. The following are the main error categories.

- *Knowledge Representation Issues* (37%) Some questions require the ability to recognize the order in which sub-events happen in a process e.g., *What is the third step of the water cycle?*. In some cases, the question only specifies one role such as the result e.g., *What process ends with a new and improved plant?*. Success in these cases depends on ability to do effective textual entailment. In other cases, certain critical information are not covered in our semantic representation. Example: _____ *is the spinning of a planet on its axis.* 'on its axis' is the critical information that differentiates *rotation* and *revolution* but unfortunately this is not covered in our semantic roles.
- *Entailment Issues* (32%) Textual entailment is noisy and provides inconsistent scores even in some simple cases. For example, the text-hypothesis pair *('all plant and animal species', 'all living things')* has an entailment score of 0.2856 while the pair *('all plant and animal species', 'plants')* has a score of 1. Some textual entailment cases are hard problems which themselves require deeper knowledge and reasoning. For example, one question requires reasoning that 'energy traveling from the sun to Earth' is related to 'energy transferred through space'. Similarly, another question requires that we know 'rub your hands together very quickly' is related to 'friction'.
- *Scoring Issues* (31%) In many cases the scoring function we use is unable to distinguish between a strong evidence that comes from one role vs. many weak partial evidences from multiple roles. Favoring cases with many weak partial evidences requires learning a threshold. We plan to investigate a learning-based approach for combining the partial evidences rather than an ad-hoc scoring function.

5 Conclusions

In this work we focused on a semantic role based representation of knowledge about processes.

We find a small set of semantic roles can be used to build an effective representation for process recognition questions. Unfortunately, automatic SRL systems require significant amounts of training data. Out-of-the-box application of a standard SRL system trained on our limited labeled data turns out to be quite noisy and doesn't yield benefits for QA. Prior work explored semi-supervised and un-supervised approaches for addressing the data sparsity issues (Fürstenau and Lapata, 2009; Lang and Lapata, 2011; Lang and Lapata, 2010). We explored a domain adaptation technique and a simple distant supervision approach. While domain adaptation yielded minor gains, we were unable to benefit from the simpler distant supervision approach. This is in part due to differences in the percentage of roles in the distant supervision and the target sentences.

Error analysis on the manual role-based QA shows representational gaps. Also, many questions require deeper reasoning that go beyond simple textual entailment. Our findings suggest the following avenues for future work: 1) Address representational gaps by a mix of pre-specified general roles and automatically discovered process specific roles, 2) Introduce additional structure within the roles to facilitate deeper reasoning, 3) Rather than relying on automatic interpretation of a handful of sentences, compose knowledge by explicitly searching for sentences that express roles in expected ways.

6 Acknowledgements

This work is funded in part by Fulbright PhD Fellowship and by the Allen Institute for Artificial Intelligence. The findings and conclusions expressed herein are those of the authors alone.

References

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, Philip Harrison, and Niranjan Balasubramanian. 2013. A study of the knowledge base requirements for passing an elementary science test. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 37–42. ACM.

Hal Daumé III. 2009. Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*.

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.

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 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1320–1331, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.