Extracting STRIPS representation of Actions for Machine Reading between the Lines

Avirup Sil, advisor: Dr Alexander Yates

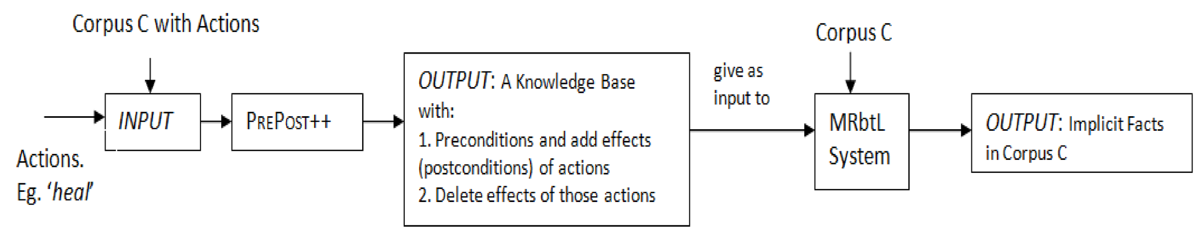
Computer and Information Sciences, Temple University

Introduction

We present:

• A novel system PREPOST++ that tackles the problem of extracting knowledge about how actions and events change the world over time.

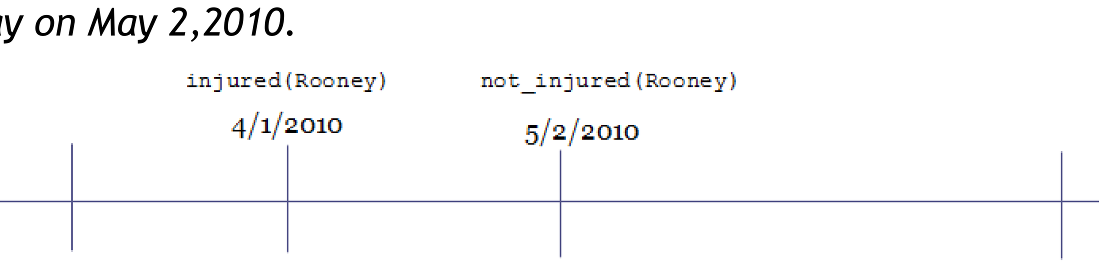
•A novel task called "Machine Reading between the Lines" (MRbtL) to measure the extraction accuracy of our extracted *knowledge base* and argue that this is a more effective evaluation technique.

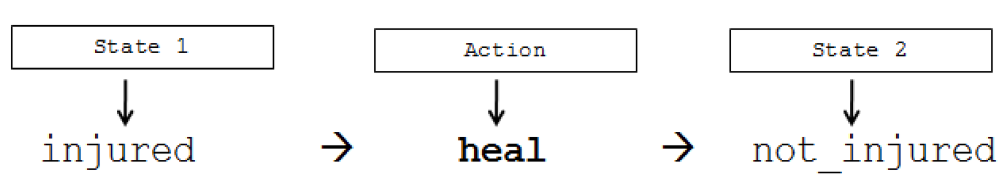


Objective and Importance of this project

The adage that "common sense isn't very common" is a truism for Artificial Intelligence (AI) and natural language understanding systems. Researchers have long recognized that a *knowledge acquisition bottle-neck* has prevented systems from acquiring and mak-ing use of the full range of common sense knowl-edge necessary for understanding and processing natural language, except in specialized domains. We explore text mining approaches to extracting common-sense knowledge about dynamics. In particular, we seek to extract knowledge about how **actions and events change the state of the world**. Consider the following text:

Eg. *Wayne Rooney was* ***injured*** *on April 1, 2010….Rooney* ***came back*** *to play on May 2,2010.*





Importance of PREPOST++ : If we ever observe a *'heal'* action in text .

- System can use this knowledge to *'read between the lines'*.
- Infer things that may never have been mentioned, such as that before the action, someone was injured, and afterwards, they were not.

Definition of Terms in our Experiments

States: Conditions of the world and its objects and attributes.
Events: Observable phenomena that occur at a particular time & place
Actions: Events brought about by rational agents
STRIPS Representation (shown in figures)consists of:

•Action
•Arguments
•Preconditions - Set of conditions for action to take place
•Postconditions/Effects - Set of conditions of how world changes after action takes place
•Add
•Delete

Our Task Definition

A STRIPS representation for action "*heal*" and "put" is as shown below:

heal
args: x,y,p,r
pre: person(x), medical_knowledge(m), has(x,m), person(y), pain(p), has(y,p)
add:
del: has(y,p)

put
args: x, y, z
pre: agent(x), possesses(x, y), empty(z)
add: on(y, z)
del: possesses(x, y), empty(z)

Our task is divided into the following:

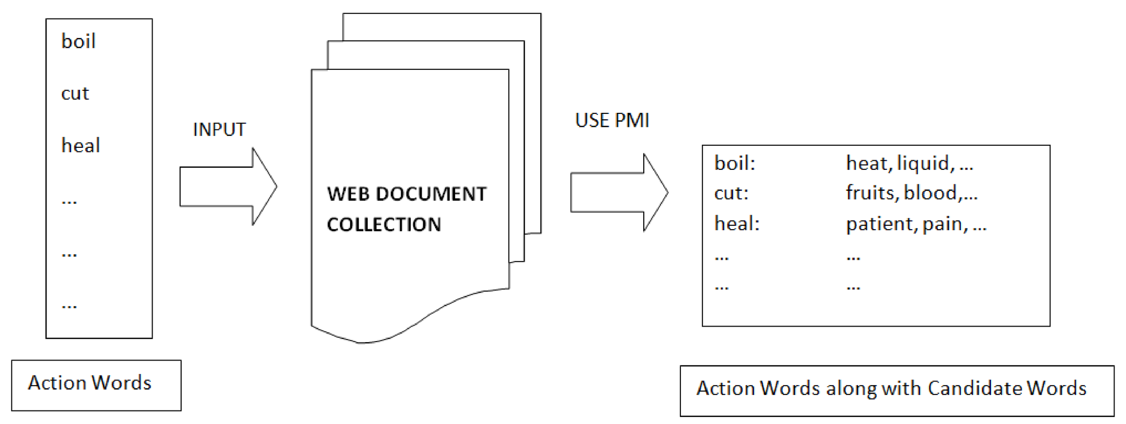
•Given a large document collection *D* as input and an event *e*, like '*heal*', we seek to achieve the above STRIPS representation.

•Once we build a Knowledge Base(KB) of these pre and postconditions of actions we evaluate the accuracy of this KB by using MRbtL.

Our Method to extract STRIPS representation

1. Collect a large Document collection for Action/Events.
   - Download all documents from web with action say 'heal'
2. Select candidate Pre/Postconditions.
   - Choose top 500 words(c) from D_c with high PMI corresponding to event *e* in document set D_e.

$$PMI(e,c) = \log \frac{|D_{\{e,c\}}|}{|D_{\{e\}}||D_{\{c\}}|}$$



- Note that here we generalize all words using Wordnet Hypernym.

3. Compute features from counts over text.
   - *Discriminator features*- We find 3-way PMI between event *e*, candidate *c* and discriminator words *f* which are words like '*before*', '*after*' etc.

$$PMI(e,c,f) = \log \frac{|D_{\{e,c,f\}}|}{|D_{\{e\}}||D_{\{c\}}||D_{\{f\}}|}$$

   - Eg. Using this we observe: *pmi*("heal", "pain", "before") > *pmi*("heal", "hospital", "before")
   - *Features from extracted relationships*- Annotate texts with structural information with a Semantic Role Labeling (SRL) system. Count how often candidate words (*c*)are associated through predicate-argument structure with certain actions(*e*).
   - Eg. "I was able to heal my chronic **pain** using all natural techniques." Here, we count how many times 'pain' occur as an argument to event 'heal'.
4. Train and test an SVM to *classify* between the true pre and postconditions extracted. Then, rank them using the prediction.

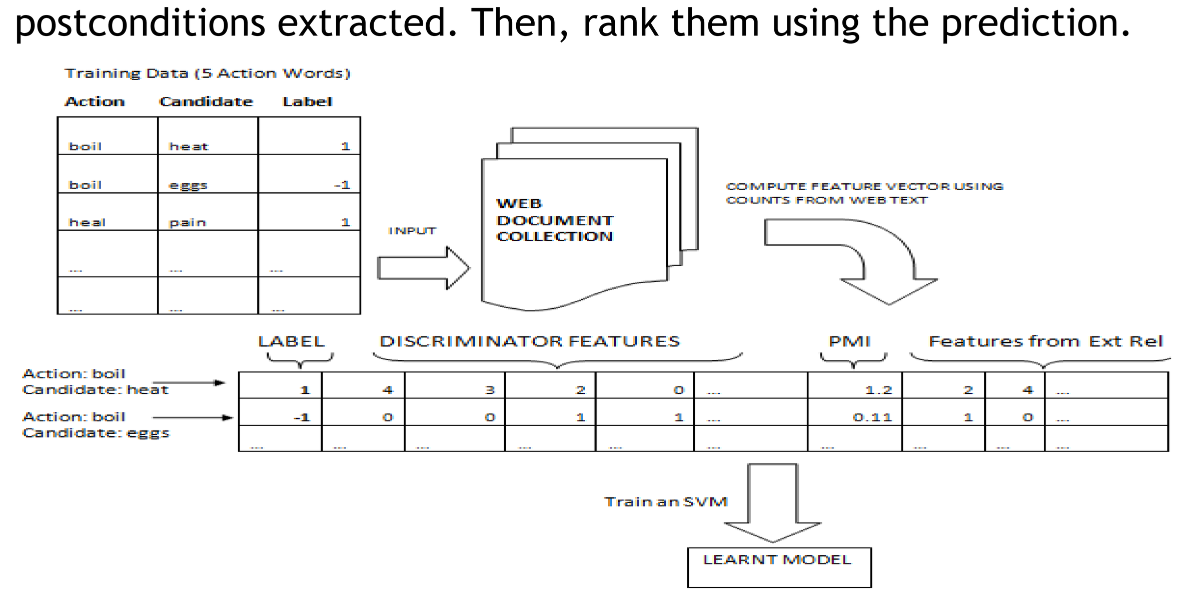


Fig1. Training of PREPOST++

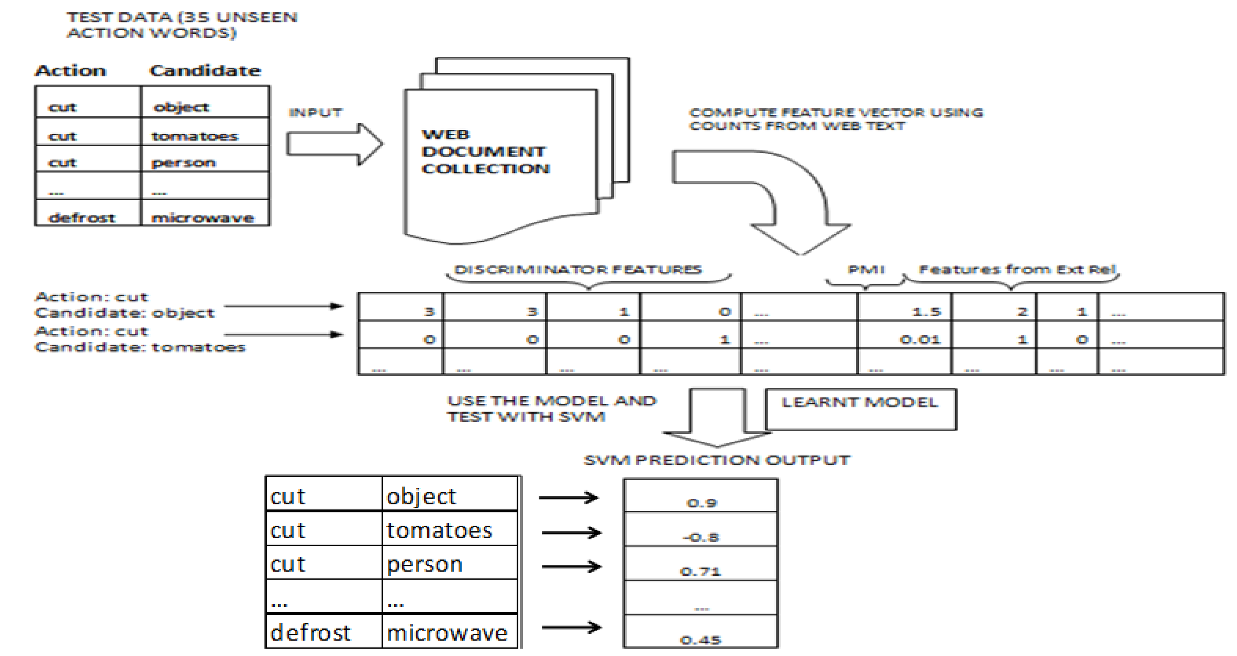


Fig2. Testing and Ranking phase of PREPOST++

Experiments:

Setup: *Input*: Action words and a set of documents containing those action words.

Output: Pre and Postconditions of those actions.

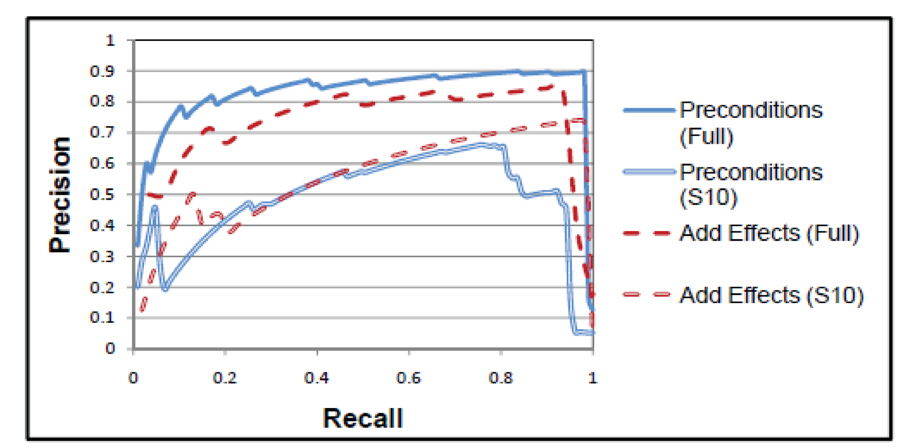


Fig3. Precision-Recall curves for extracting preconditions and postconditions/add effects

Technique	Prec.	Recall	F1
All preds. have same var.	32	33	32
Each pred. has distinct var.	56	58	57
Semantic role heuristic	**73**	**72**	**72**

Table 1. Precision and recall of our complete representation with extracted predicates & arguments

Machine Reading between the Lines

Given a KB and a corpus C from which a *Machine Reading* or information extraction system has extracted a database of relational extractions RE, a system for the task of MRbtL produces additional ground extractions, which we call *implicit extractions* IE, which can be logically deduced from KB ∧ RE. This setup is then judged against:

•Precision

•Amount and

•Redundancy

If IE is accurate and contains many nonredundant extractions not obvious *a priori*), then we judge KB to be a 'useful' knowledge base.

Building an MRbtL system: We use the set of predicates and their arguments discovered by our SRL as our explicit relational extractions RE. We then apply the following axioms:

$$\forall_{\mathbf{args}} e(\mathbf{args}, t_2) \Rightarrow p(\mathbf{args}_p, t_1)$$
$$\forall_{\mathbf{args}} e(\mathbf{args}, t_2) \Rightarrow a(\mathbf{args}_a, t_3)$$
$$\forall_{\mathbf{args}} e(\mathbf{args}, t_2) \Rightarrow \neg d(\mathbf{args}_d, t_3)$$

Here, $args_x$ represents the subset of the arguments to which the predicate x applies.

Eg. Our SRL discovers the predicate *draining(a0, a1)* ∧ *the acid solution(a1)* from the sentence, *"This is done by inverting the battery and draining the acid solution out the vent holes in the battery cover."*

By applying the extracted precondition that the second argument of a '*draining*' event must be a '*liquid*', we can infer that *liquid(a1)* is true immediately before the event. Eg. If we had inferred *solution*(a1), that would be considered redundant.

MRbtl Experiments: We measured the quality of our implicit extractions by taking a random samples of 100, and having two judges classify each extraction for correctness and redundancy in the context of the sentence and document from which it was extracted.

Eg. From the sentence *"When a sharp object, like a fingernail or thorn, scrapes along your skin . . . "*, our MRbtL system extracted that the *fingernail* is an *object*, since the instrument of a scraping action needs to be an *object*. Both annotators considered this extraction correct, but redundant, since the sentence explicitly mentions that a *fingernail* is a kind of *object*.

	PREPOST	PREPOST++	*k*	signif.
accur.	45%	73%	0.65	p<0.01
redun.	21%	12%	0.91	p = .13
num.	54,300	67,192	N/A	N/A

Table 2. The KB extracted by PREPOST++ can identify more, and more accurate, implicit extractions than PREPOST's KB, and fewer implicit extractions overlap with explicit extractions. The first two columns record the accuracy and redundancy (averaged over two annotators on sample of 100), and total number of implicit extractions. *k* indicates Cohen's *k* inter-annotator agreement score, and *p*-values for the significance tests are calculated using a two-sided Fisher's exact test.

Conclusion and Future Work

We have achieved the following:

•A system for extracting a complete STRIPS representation of 40 common actions from text, with an overall F1 of 0.72.

•The extracted knowledge base can be used to accurately identify information in a document that is never stated explicitly.

•We call this evaluation scenario *"Machine Reading between the Lines."*

Future directions include:

•Extracting more sophisticated representations of action semantics.

•To incorporate extracted implicit information into applications like information retrieval.

References

•Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In Proceedings of *ACL-IJCNLP 2009*.

•Nello Cristianini and John Shawe-Taylor. 2000. An Introduction to Support Vector Machines(SVM) and other kernel-based learning methods. Cambridge University Press.

•Sil, A.; Huang, F.; and Yates, A. 2010. Extracting action and event semantics from web text. In *AAAI Fall Symposium on Common-Sense Knowledge (CSK)*.

•Schank, R., and Abelson, R. 1977. Scripts, plans, goals and understanding: an inquiry into human knowledge structures. Erlbaum.