Exploiting Partially Annotated Data for Temporal Relation Extraction

Qiang (John) Ning, Chuchu Fan
Department of Electrical and Computer Engineering
University of Illinois

Zhongzhi (Mark) Yu
Department of Computer Science
University of Illinois

Dan Roth
Department of Computer Science
University of Pennsylvania
University of Illinois

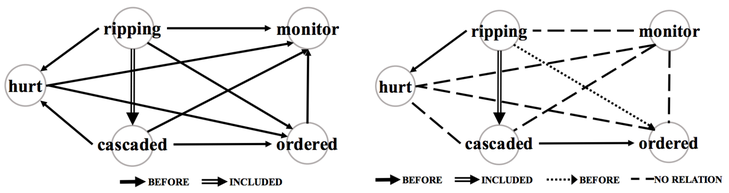
Introduction

Extracting temporal relations (TempRel) between events (e.g., before, after, includes, equal) is an important task in natural language understanding.

The TempRels in a doc can be conveniently modeled as a graph:

- Node = Event
- Edge = TempRel

Example: . . . tons of earth **cascaded** down a hillside, **ripping** two houses from their foundations. No one was **hurt**, but firefighters **ordered** the evacuation of nearby homes and said they'll **monitor** the shifting ground.. . .



TempRel annotation requires labeling all the edges, which is very labor intensive.

- Annotating each edge is time-consuming
- Too many edges! $O(n^2)$

As a result, only a small number of docs are fully annotated. Less docs = Less coverage of phenomena!

Possible solution:

- Collect more documents
- **Learn from partial annotation (this paper)**

Data

- F: 36 fully annotated docs from TBDense
- P: 220 partially annotated docs from TBAQ

Train: 22 F docs + 220 P docs
Dev: 5 F docs
Test: 9 F docs

Data	#Doc	#Edges	Ratio	Type
TB-Dense	36	6.5K	100%	$\mathcal{F}$
TBAQ	220	2.7K	12%	$\mathcal{P}$

Table 1: Corpus statistics of the fully and partially annotated dataset used in this work. TBAQ: The union of TimeBank and AQUAINT, which is the training set provided by the TempEval3 workshop. #Edges: The number of annotated edges. Ratio: The proportion of annotated edges.

How to Make Use of Partially Annotated Data

Algorithm 1: Joint learning from $\mathcal{F}$ and $\mathcal{P}$ by bootstrapping

Input: $\mathcal{F}$, $\mathcal{P}$, Learn, Inference

1 $S_{\mathcal{F}} = \text{Learn}(\mathcal{F})$
2 Initialize $S_{\mathcal{F}+\mathcal{P}} = S_{\mathcal{F}}$
3 **while** *convergence criteria not satisfied* **do**
4 $\quad \tilde{\mathcal{P}} = \emptyset$
5 $\quad$ **foreach** $p \in \mathcal{P}$ **do**
6 $\quad\quad \hat{\mathbf{y}} = \text{Inference}(p; S_{\mathcal{F}+\mathcal{P}})$
7 $\quad\quad \tilde{\mathcal{P}} = \tilde{\mathcal{P}} \cup \{(\mathbf{x}, \hat{\mathbf{y}})\}$
8 $\quad S_{\mathcal{F}+\mathcal{P}} = \text{Learn}(\mathcal{F} + \tilde{\mathcal{P}})$
9 **return** $S_{\mathcal{F}+\mathcal{P}}$

CoDL Framework
(Chang et al., 2012.)

$$\hat{\mathcal{I}} = \underset{\mathcal{I}}{\text{argmax}} \sum_{i<j} \sum_{r \in \mathcal{R}} f_r(ij)\mathcal{I}_r(ij)$$
$$\text{s.t.} \quad \Sigma_r \mathcal{I}_r(ij) = 1, \quad \text{(uniqueness)}$$
$$\mathcal{I}_{r_1}(ij) + \mathcal{I}_{r_2}(jk) - \Sigma_{m=1}^{N} \mathcal{I}_{r_3^m}(ik) \leq 1, \quad \text{(transitivity)}$$

Three components to keep in mind:

- Bootstrapping: New annotations are predicted on P
- Structural constraints: Enforced via ILP constraints
- Partial annotation: Enforced via equality constraints

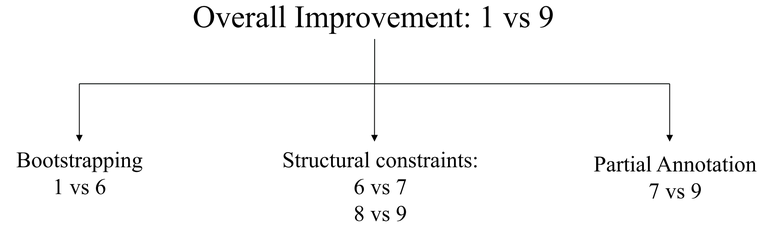
Benchmark Performance on the Test Split of TimeBank-Dense

No.	Training		*Same Sentence*			*Nearby Sentence*			**Overall**			**Awareness**		
	Data	Bootstrap	P	R	F	P	R	F	P	R	F	P	R	F
1	$\mathcal{F}$	-	47.1	49.7	48.4	40.2	37.9	39.0	**42.1**	41.0	**41.5**	**40.0**	40.7	**40.3**
2	$\mathcal{P}^{Full}$	-	37.0	33.1	35.0	34.4	19.6	24.9	37.7	23.6	29.0	36.9	24.0	29.1
3	$\mathcal{P}$	-	34.1	52.5	41.3	26.1	48.1	33.8	30.2	**52.1**	38.2	28.6	**49.9**	36.4
4	$\mathcal{F}$+$\mathcal{P}^{Full}$	-	38.5	32.2	35.1	40.1	38.1	39.1	40.8	35.3	37.8	37.1	36.2	36.6
5	$\mathcal{F}$+$\mathcal{P}$	-	43.7	43.9	43.8	39.1	38.3	38.7	41.8	40.7	41.2	38.6	41.4	40.0
6	$\mathcal{F}$+$\mathcal{P}^{Empty}$	Local	41.7	50.3	45.6	39.5	48.1	43.4	41.8	50.4	45.7	40.9	47.5	43.9
7	$\mathcal{F}$+$\mathcal{P}^{Empty}$	Global	44.7	55.5	49.5	40.1	48.7	44	**42.0**	**51.4**	**46.2**	**41.1**	**48.3**	**44.4**
8	$\mathcal{F}$+$\mathcal{P}$	Local	43.6	50	46.6	43	46.9	44.8	43.7	47.8	45.6	42	45.6	43.7
9	$\mathcal{F}$+$\mathcal{P}$	Global	44.9	56.1	49.9	43.4	52.3	47.5	**44.7**	**54.1**	**49.0**	**44.1**	**50.8**	**47.2**

P^{Full}: P with missing annotations filled by "vague"
P^{Empty}: P with all annotations removed
Bootstrap: referring to specific implementations of Line 6 in Algorithm 1.

- Local=don't enforce structural constraints.
- Global=enforce structural constraints.

Overall Improvement: 1 vs 9



Discussion

Machine Learning requires supervision, but task specific annotation is significantly limited by expertise and cost. This raises three key questions:

- How can we learn from imperfect supervision, e.g., partial, noisy, or indirect? **(Answered by this paper)**
- How can we characterize the improvement from bootstrapping, structural constraints and partial data?
- What is the implication of structured data on annotation?

Conclusion

TempRel annotation is labor intensive. Fully annotated datasets (F) are relatively small and there are more partial datasets (P).This work first investigates learning from both types of datasets, and shows promise, which is a good starting point for further investigations of incidental supervision and data collection schemes, of the TempRel extraction task and of other general machine learning tasks.

[0] D. Hovy and E. Hovy. NAACL'12. Exploiting Partial Annotations with EM Training. [1] M. Chang et al. 2012. Structured learning with constrained conditional models. [2] N. UzZaman et al. SemEval'13. Task 1: TempEval-3: Evaluating time expressions, events, and temporal relations. [3] T. Cassidy et al. ACL'14. An annotation framework for dense event ordering. [4] D. Roth. AAAI'17. Incidental supervision: Moving beyond supervised learning. [5] Q. Ning et al. EMNLP'17. A structured learning approach to temporal relation extraction.

ILLINOIS
Cognitive Computation Group
Penn Engineering