# A STRUCTURED LEARNING APPROACH TO TEMPORAL RELATION EXTRACTION

**Qiang Ning**, Zhili Feng, Dan Roth

Computer Science
University of Illinois, Urbana-Champaign
&
University of Pennsylvania

# Towards Natural Language Understanding

1. 
2. 
3. 
4. 

.....

.....

11. Reasoning with respect to Time

# Understanding Time in Text

- Understanding time is key to understanding events
  - Timeline construction (e.g., news stories, clinical records), time-slot filling, Q&A, causality analysis, pattern discovery, etc.
- Applications depend on two fundamental tasks
  - Time expression extraction and normalization
    - "yesterday"→2017-09-09
    - "Thursday after labor day" → 2017-08-31
    - 2 time expressions in every 100 tokens (in TempEval3 datasets)

"Time" that is expressed **explicitly**

  - Temporal relation extraction

"Time" that is expressed **implicitly**

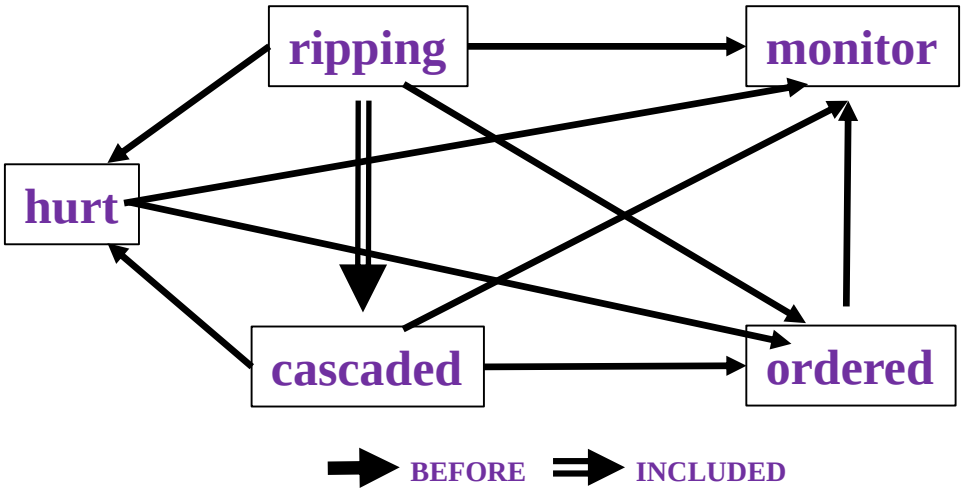
    - "A" happens BEFORE/AFTER "B"
    - 12 temporal relations in every 100 tokens (in TempEval3 datasets)

# Graph Representation of Temporal Relations

- … In Los Angeles that lesson was brought home today when 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 until March 23rd.

Five Relation types:

Before; After; Include; Included; equal

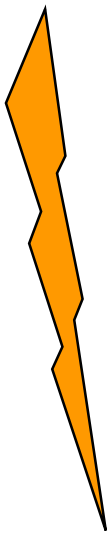


# CHALLENGE I: STRUCTURE

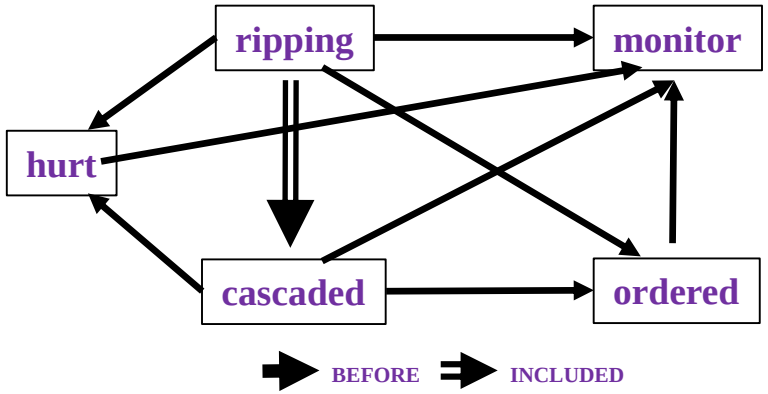
- Structure of a temporal graph [Bramsen et al.'06; Chambers & Jurafsky'08l Do et. al.'12]
  - Symmetry: "A BEFORE B"→"B AFTER A"
  - Transitivity: "A BEFORE B" + "B BEFORE C"→"A BEFORE C"
  - Relations are highly interrelated, but existing methods learn models by considering a single pair at a time.

**Existing methods**

"ripping" vs "hurt"
"ripping" vs "cascaded"
"ripping" vs "monitor"
…



**Expectation**

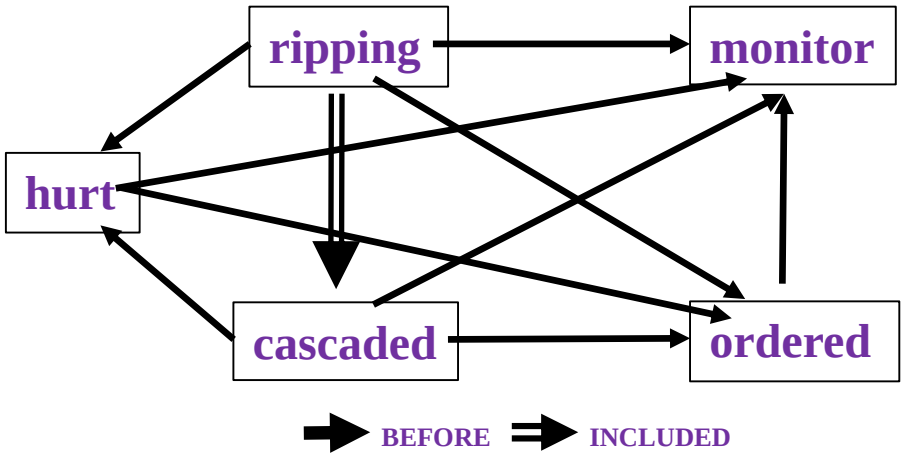


# CHALLENGE II: MISSING RELATIONS

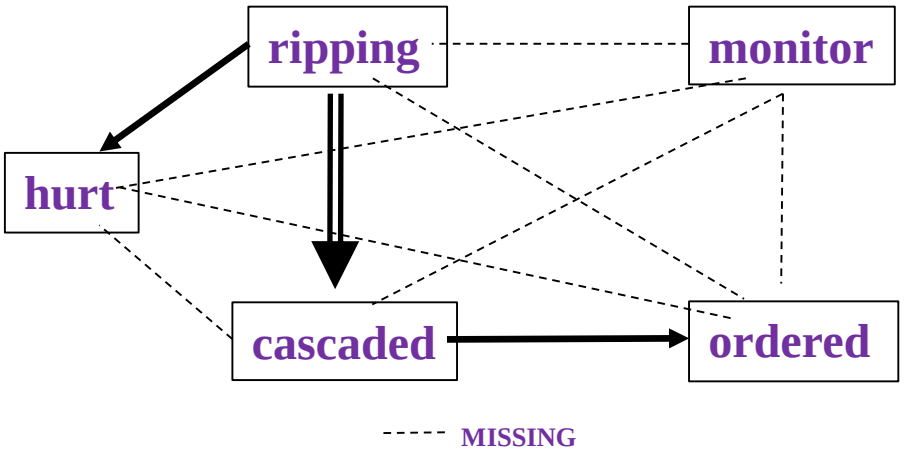
- Most of the relations are left unannotated

- Problems of existing approaches
- Addressing both challenges
  - Structured Prediction
  - Dealing with missing relations in the annotation.

**Ground Truth**




**Provided Annotation (TempEval3)**

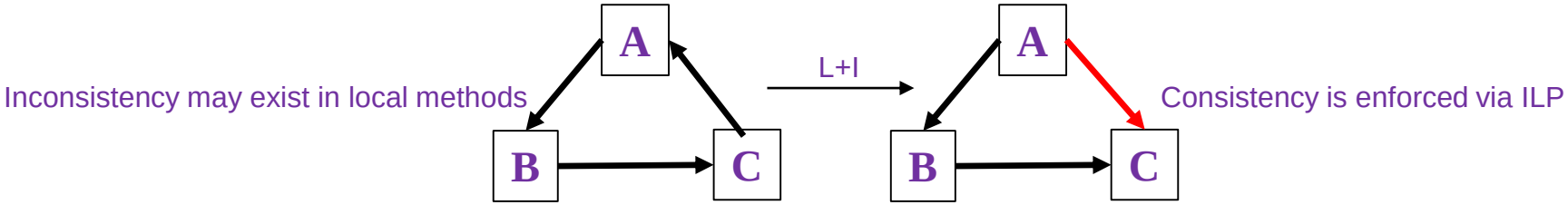



- Missing relations arise in three scenarios:
  - The annotators did not look at a pair of events (e.g, long distance)
  - The annotators could not decide among multiple options
  - Annotators' disagreements
- The annotation task is **difficult** if done at a single event pair level

# Existing Approaches


[1] Mani et al., ACL2006
[2] Chambers et al., ACL2007
[3] Bethard, ClearTK-TimeML: TempEval 2013
[4] Laokulrat et al., SEM2013
[5] Bramsen et al., EMNLP2006
[6] Chambers and Jurafsky, EMNLP2008
[7] Do et al., EMNLP2012


- Local methods [1-4]
  - Learn models or design rules that make pairwise decisions between each pair of events
  - Global consistency (i.e., symmetry and transitivity) is not enforced




- Local methods + Global Inference (L+I) [5-7]
  - Formulate the problem as an integer linear programming (ILP) over the entire graph, on top of pre-learnt local models
  - Consistency guaranteed: structural requirements are added as declarative constraints to the ILP
  - Performance improved: Local decisions may be corrected via global consideration

# Challenge I: Consistent Decision Making Is not Sufficient

- Neither local methods nor L+I methods account for structural constraints in the learning phase.
- But information from other events is often necessary.

tons of earth cascaded down a hillside,

- …**ripping** two houses…firefighters **ordered** the evacuation of nearby homes… (What's the temporal relation between ripping and ordered? It's difficult to tell.)
  - As a result, (ripping, ordered)=BEFORE cannot be supported given the local information, resulting in overfitting. .
- However, observing that (ripping, ordered)=BEFORE actually **results from** (ripping, cascaded)=INCLUDED and (cascaded, ordered)=BEFORE, rather than the local text itself, supports better learning.

# Proposed Approach: Inference-Based Training

**Local Training (Perceptron)**

For each $(x, y)$

$$\hat{y} = sgn(w^T x)$$

If $y \neq \hat{y}$

Update $w$

- $(x, y)$: feature and label for a single pair of events
- When learning from $(x, y)$, the algorithm is unaware of decisions with respect to other pairs.

**IBT (Structured Perceptron)**

For each $(X, Y)$

$$\hat{Y} = \underset{Y \in \mathcal{C}}{\operatorname{argmax}}\, W^T X$$

If $Y \neq \hat{Y}$

Update $W$

- $(X, Y)$: features and labels from a whole document
- $Y \in \mathcal{C}$: Enforce consistency through constraint $\mathcal{C}$.

# Proposed Approach: Inference-Based Training

- Inference step
  - $\mathcal{E}$ Event node set, $\mathcal{Y}$ temporal label set
  - $\boldsymbol{I_r(ij)}$ Boolean variable for event pair (i,j) being relation r
  - $\boldsymbol{f_r(ij)}$ softmax score of event pair (i,j) being relation r
  - $r_m$ temporal relations implied by $r_1$ and $r_2$

$$\hat{I} = arg\min_{I} \sum_{ij \in \mathcal{E}} \sum_{r \in \mathcal{Y}} f_r(ij) I_r(ij)$$

s.t. $\forall i, j, k \in \mathcal{E}$

$$\sum_{r} I_r(ij) = 1$$ Uniqueness

$$I_r(ij) = I_{\neg r}(ji)$$ Symmetry

$$I_{r1}(ij) + I_{r2}(jk) - \sum_{m} I_{rm}(ik) \leq 1$$ Generalized Transitivity

# PROPOSED APPROACH: INFERENCE-BASED TRAINING

- Constraint-Driven Learning
  - Make use of unannotated data

**Algorithm 2:** Constraint-driven learning algorithm

**Input:** Labeled set $\mathcal{L}$, unlabeled set $\mathcal{U}$, weighting coefficient $\gamma$

1 Perform closure on each graph in $\mathcal{L}$
2 Initialize $\mathbf{w}_r = \text{Learn}(\mathcal{L})_r, \forall\, r \in \mathcal{Y}$
3 **while** *convergence criteria not satisfied* **do**
4  $\mathcal{T} = \emptyset$
5  **foreach** $\mathbf{x} \in \mathcal{U}$ **do**
6  $\hat{\mathbf{y}} = \arg\max_{\mathbf{y} \in \mathcal{C}} f(\mathbf{x}, \mathbf{y})$
7  Perform graph closure on $\hat{\mathbf{y}}$
8  $\mathcal{T} = \mathcal{T} \cup \{(\mathbf{x}, \hat{\mathbf{y}})\}$
9  $\mathbf{w}_r = \gamma \mathbf{w}_r + (1 - \gamma)\text{Learn}(\mathcal{T})_r, \forall\, r \in \mathcal{Y}$
10 **return** $\{\mathbf{w}_r\}_{r \in \mathcal{Y}}$

Chang et al., Guiding semi-supervision with constraint-driven learning. ACL2007.
Chang et al., Structured learning with constrained conditional models. Machine Learning 2012.

# Results (Challenge I)

- When gold related pairs are known (TE3, Task C, Relation only)

Enforcing constraints only at decision time.

Enforcing constraints during learning

| System | Method | Precision | Recall | F1 |
|---|---|---|---|---|
| UTTime [1] | Local | 55.6 | 57.4 | **56.5** |
| AP | Local | 58.0 | 55.3 | 56.6 |
| AP+ILP | L+I | 62.2 | 61.1 | 61.6 |
| SP+ILP | S+I | 69.1 | 65.5 | **67.2** |

[1] Laokulrat et al., UTTime: Temporal relation classification using deep syntactic features, SEM2013

# HOWEVER, REALISTICALLY

- When gold related pairs are NOT known (TE3, Task C)

| System | Method | Precision | Recall | F1 |
|---|---|---|---|---|
| ClearTK [1] | Local | 37.2 | 33.1 | 35.1 |
| AP | Local | 35.3 | 37.1 | 36.1 |
| AP+ILP | L+I | 35.7 | 35.0 | 35.3 |
| SP+ILP | S+I | 32.4 | 45.2 | 37.7 |

- Performance drops significantly.
- Structured learning is not helping as much as previously in the presence of **missing**, **vague relations**
- Existing methods of handling vague relations are ineffective:
  - Simply add "vague" to the temporal label set
  - Train a classifier or design rules for "vague" vs. "non-vague"

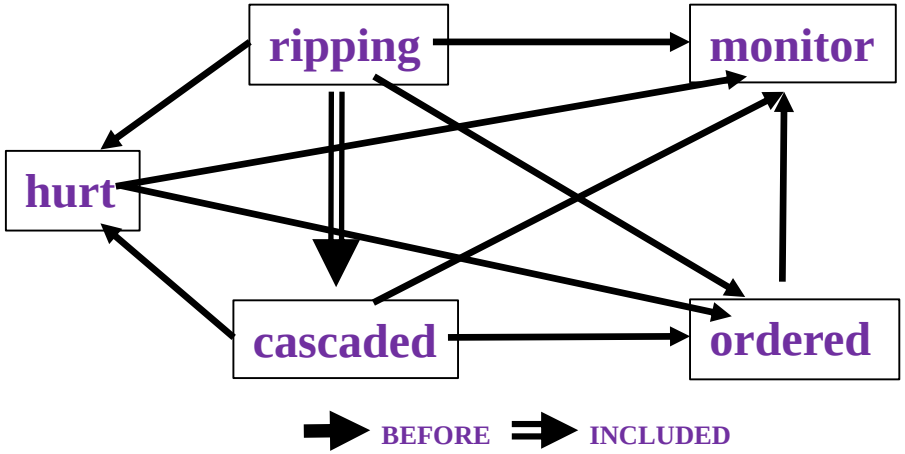
[1] Bethard, ClearTK-TimeML: A minimalist approach to TempEval 2013

# Challenge II: Missing Relations

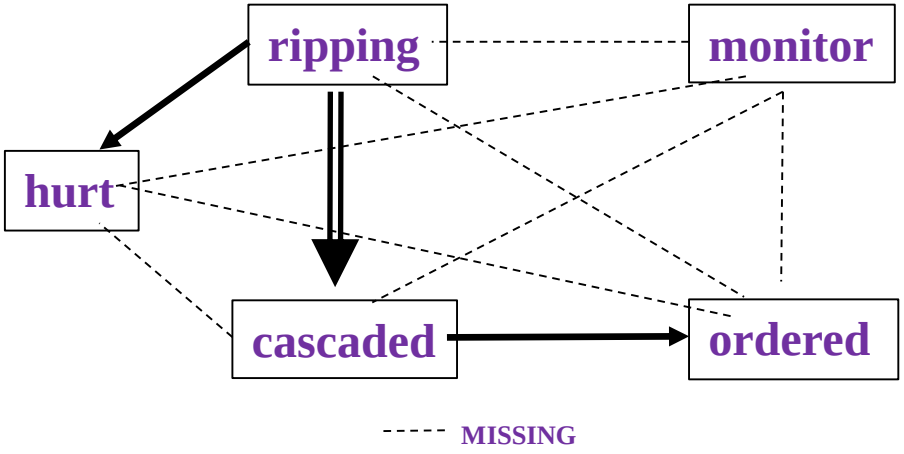
| Type | | #TLINK | % |
|---|---|---|---|
| Annotated | | 582 | 1.8 |
| Missing | Inferred | 2840 | 8.7 |
| | Unknown | 29240 | 89.5 |

- Most of the relations are left unannotated

**Ground Truth**




**Provided Annotation (TempEval3)**




- The annotation task is **difficult** if done at a single event pair level
- Some of the missing relations can be **inferred**
  - Saturate the graph via symmetry and transitivity
- The vast majority, cannot

## HANDLING VAGUE RELATIONS

- 1. Ignore vague labels during training
  - Many vague pairs are not really vague but rather pairs that the annotators failed to look at.
  - The imbalance between vague and non-vague relations makes it hard to learn a good vague classifier.
  - **The Vague relation is fundamentally different** from other relation types.
    - If (A, B) = BEFORE, then it's always BEFORE regardless of other events.
    - But if (A, B) = VAGUE, the relation can change if more context is provided.
- 2. Apply post-filtering using KL divergence
  - For each pair, we have a predicted distribution over possible relations.
  - Compute the KL divergence of this distribution with the uniform distribution, and filter out predictions that have a low score.
  - $\delta_i = \sum_{m=1}^{M} f_{rm}(i) \log(M f_{rm}(i))$, M=#labels, $f_r(i)$ =score for pair $i$.
  - High similarity to the uniform distribution, $\delta_i < \tau$, implies unconfident prediction→ change decision to Vague.

# RESULTS (CHALLENGE II)

- When gold related pairs are NOT known (TE3, Task C)
- Apply the post-filtering method proposed above

| System | Method | Precision | Recall | F1 |
|---|---|---|---|---|
| ClearTK [1] | Local | 37.2 | 33.1 | **35.1** |
| AP | Local | 35.3 | 37.1 | 36.1 |
| AP+ILP | L+I | 35.7 | 35.0 | 35.3 |
| SP+ILP | S+I | 32.4 | 45.2 | **37.7** |

[1] Bethard, ClearTK-TimeML: A minimalist approach to TempEval 2013

# Overall Results

- TempEval3 dataset is known to suffer from TLINK sparsity issues.
- **Timebank-dense** is another dataset with much denser TLINK annotations.
- Significant improvement over CAEVO, the previously best system on Timebank-dense.

| System | Method | Precision | Recall | F1 |
|---|---|---|---|---|
| ClearTK [1] | Local | 46.04 | 20.90 | 28.74 |
| CAEVO [2] | L+I | 54.17 | 39.49 | **45.68** |
| SP+ILP | S+I | 45.34 | 48.68 | **46.95** |
| CoDL+ILP | S+I | 45.57 | 51.89 | **48.53** |

[1] Bethard, ClearTK-TimeML: A minimalist approach to TempEval 2013
[2] Chambers et al., Dense event ordering with a multi-pass architecture. TACL 2014

# CONCLUSION

Thanks

- Identifying Temporal relations between events is a highly structured task
  - This results also in low quality annotation (vague relations)

- This work shows that
  - Using structured information during learning is important
  - The structure can be exploited in an unsupervised way (via CoDL) to further improve results
  - Vagueness is the result of lack of information rather than a concrete relation. KL-driven post-filtering is shown to be an effective way to treat vague relations.

- A lot more work is needed on temporal reasoning from text