ACL2020

A Joint Neural Model for Information Extraction with Global Features

Ying Lin[1], **Heng Ji**[1], **Fei Huang**[2], **Lingfei Wu**[3]
[1]University of Illinois at Urbana-Champaign
[2]Alibaba DAMO Academy [3]IBM Research
{yinglin8,hengji}@illinois.edu,
f.huang@alibaba-inc.com, wuli@us.ibm.com

Main Work

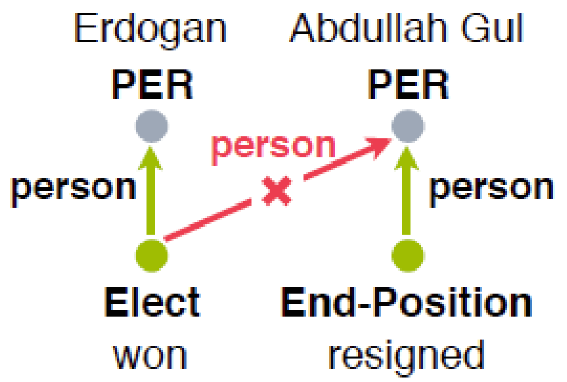
- Most existing joint neural models for Information Extraction (IE) use local task-specific classifiers to predict labels for individual instances **regardless of their interactions**

- we propose a joint neural framework, ONEIE, that aims to extract the globally optimal IE result as a graph from an input sentence.

- Experiments show that adding global features improves the performance of our model and **achieves new state-of-the-art on all subtasks**.

Task

- **Information Extraction** (IE) aims to extract structured information from unstructured texts.
  - It is a **complex task** comprised of a wide range of subtasks, such as named, nominal, and pronominal mention extraction, entity linking, entity coreference resolution, relation extraction, event extraction, and event coreference resolution.



Example: Prime Minister **Abdullah Gul** *resigned* earlier Tuesday to make way for **Erdogan**, who *won* a parliamentary seat in by-elections Sunday.

Model

- our ONEIE framework perform entity, relation, and event extraction within a unified framework
  - Entity Extraction
  - Relation Extraction
  - Event Extraction

Model

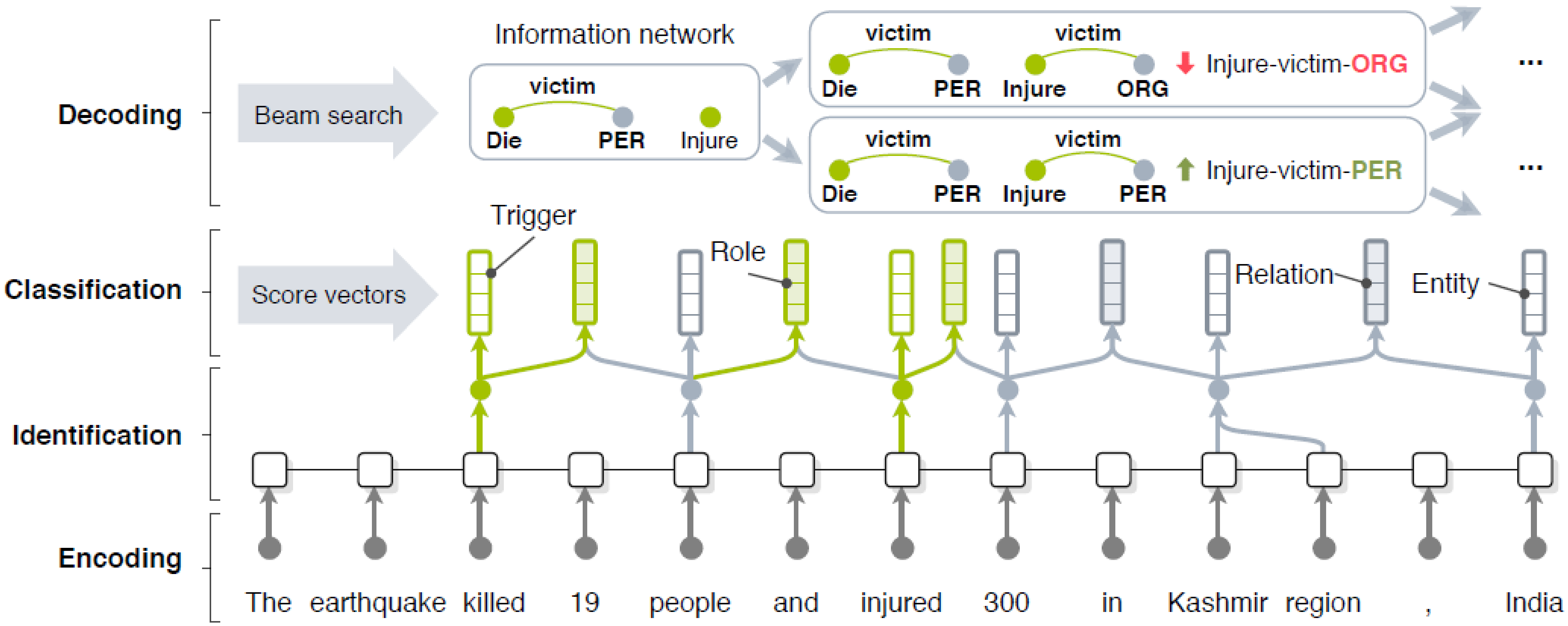


Figure 2: An illustration of our end-to-end joint information extraction framework OneIE at the test stage. We do not show all pairwise links for simplicity purposes.

Model

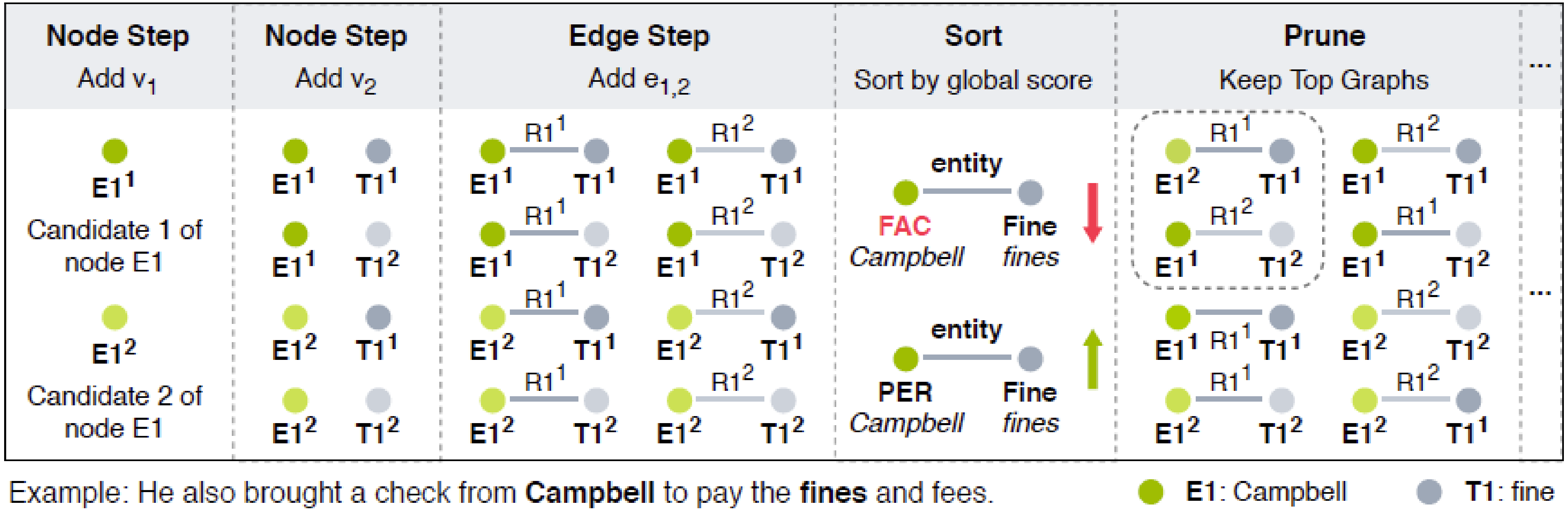


Figure 4: An illustration of our decoding algorithm. At each step, we expand each candidate graph by adding a new node and possible edges between it and existing nodes. After that, we rank all expanded graphs and keep the top ones.

Global Feature

- A limitation of local classifiers is that they are incapable of capturing inter-dependencies between knowledge elements in an information network.We consider two types of inter-dependencies in our framework.
  - The first type of inter-dependency is **Cross-subtask-interactions** between entities, relations,and events.

  Example:

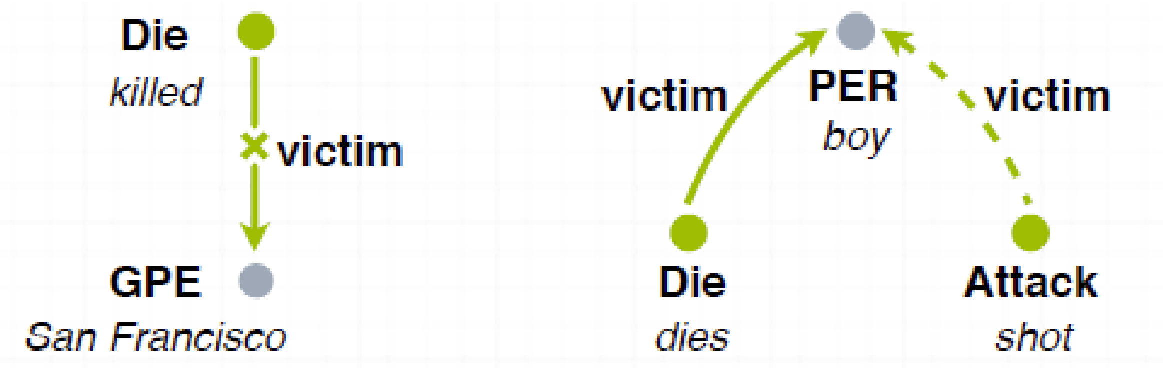
  A civilian aid worker from **San Francisco** was **killed** in an attack in Afghanistan

  - Another type of inter-dependency is Crossinstance interactions between multiple event and/or relation instances in the sentence.

  Example:

  South Carolina **boy**, 9, **dies** during hunting trip after his father accidentally **shot** him on Thanksgiving Day

Global Feature



(a) Cross-subtask Interaction (b) Cross-instance Interactions

Figure 3: Examples of inter-dependencies between elements in information networks. (a) A VICTIM edge is unlikely to exist between a GPE entity and a DIE event trigger. (b) The VICTIM of a DIE event is likely to be the VICTIM of an ATTACK event in the same sentence.

Global Feature

Categary	Description
Role	1. The number of entities that act as <role$_i$> and <role$_j$> arguments at the same time.
	2. The number of <event_type$_i$> events with <number> <role$_j$> arguments.
	3. The number of occurrences of <event_type$_i$>, <role$_j$>, and <entity_type$_k$> combination.
	4. The number of events that have multiple <role$_i$> arguments.
	5. The number of entities that act as a <role$_i$> argument of an <event_type$_j$> event and a <role$_k$> argument of an <event_type$_l$> event at the same time.
Relation	6. The number of occurrences of <entity_type$_i$>, <entity_type$_j$>, and <relation_type$_k$> combination.
	7. The number of occurrences of <entity_type$_i$> and <relation_type$_j$> combination.
	8. The number of occurrences of a <relation_type$_i$> relation between a <role$_j$> argument and a <role$_k$> argument of the same event.
	9. The number of entities that have a <relation_type$_i$> relation with multiple entities.
	10. The number of entities involving in <relation_type$_i$> and <relation_type$_j$> relations simultaneously.
Trigger	11. Whether a graph contains more than one <event_type$_i$> event.

Table 1: Global feature categories.

Global Feature

If we ignore the inter-dependencies between nodes and edges, we can simply predict the label with the highest score for each knowledge element and thus generate the locally best graph $\hat{G}$. The score of $\hat{G}$ can be calculated as

$$s'(\hat{G}) = \sum_{t \in T} \sum_{i=1}^{N^t} \max \hat{\boldsymbol{y}}_i^t,$$

weight of each feature during training. Given a graph G, we represent its global feature vector as $\boldsymbol{f}_G = \{f_1(G), ..., f_M(G)\}$, where M is the number of global features and $f_i(\cdot)$ is a function that evaluates a certain feature and returns a scalar. For example,

$$f_i(G) = \begin{cases} 1, & G \text{ has multiple ATTACK events} \\ 0, & \text{otherwise.} \end{cases}$$

Next, ONEIE learns a weight vector $\boldsymbol{u} \in \mathbb{R}^M$ and calculates the global feature score of G as the dot product of $\boldsymbol{f}_G$ and $\boldsymbol{u}$. We define the global score of G as the sum of its local score and global feature score, namely

$$s(G) = s'(G) + \boldsymbol{u}\boldsymbol{f}_G,$$

We make the assumption that the gold-standard graph for a sentence should achieve the highest global score. Therefore, we minimize the following loss function

$$\mathcal{L}^{\mathrm{G}} = s(\hat{G}) - s(G),$$

Dataset

- **ACE05-R**
  - includes named entity and relation annotations
- **ACE05-E**
  - includes entity, relation, and event annotations
  - We keep 7 entity types, 6 coarsegrained relation types, 33 event types, and 22 argument roles.
- **ACE05-E+**
  - adding back the order of relation arguments, pronouns, and multi-token event triggers, which have been largely ignored in previous work.
- **ERE-EN**
  - from the Entities, Relations and Events(ERE) annotation task created under the Deep Explorationand Filtering of Test (DEFT) program
  - LDC2015E29,LDC2015E68, and LDC2015E78

Result

Dataset	Task	DYGIE++	BASELINE	ONEIE
ACE05-R	Entity	88.6	-	**88.8**
	Relation	63.4	-	**67.5**
ACE05-E	Entity	89.7	**90.2**	**90.2**
	Trig-I	-	76.6	**78.2**
	Trig-C	69.7	73.5	**74.7**
	Arg-I	53.0	56.4	**59.2**
	Arg-C	48.8	53.9	**56.8**

Table 3: Results on ACE2005 datasets (F-score, %).

Dataset	Task	DYGIE++*	ONEIE*
ACE05-E	Entity	**90.7**	90.3
	Trig-I	76.5	**78.6**
	Trig-C	73.6	**75.2**
	Arg-I	55.4	**60.7**
	Arg-C	52.5	**58.6**

Table 4: Experiment results on ACE05-E (F-score, %). DYGIE++* and ONEIE* use a four-model ensemble optimized for trigger detection.

Task	Entity	Trig-I	Trig-C	Arg-I	Arg-C	Relation
ACE05-E$^+$	89.6	75.6	72.8	57.3	54.8	58.6
ERE-EN	87.0	68.4	57.0	50.1	46.5	53.2

Table 5: New benchmark results (F-score, %).

Dataset	Training	Entity	Relation	Trig-C	Arg-C
ACE05-CN	CN	88.5	62.4	65.6	52.0
	CN+EN	89.8	62.9	67.7	53.2
ERE-ES	ES	81.3	48.1	56.8	40.3
	ES+EN	81.8	52.9	59.1	42.3

Table 7: Results on ACE05-CN and ERE-ES (F-score, %). For ACE05-CN, EN refers to ACE05-E$^+$. For ERE-ES, EN refers to ERE-EN.

Result

	Positive Feature	Weight
1	A TRANSPORT event has only one DESTINATION argument	2.61
2	An ATTACK event has only one PLACE argument	2.31
3	A TRANSPORT event has only one ORIGIN argument	2.01
4	An END-POSITION event has only one PERSON argument	1.51
5	A PER-SOC relation exists between two PER entities	1.08
6	A GEN-AFF relation exists between ORG and LOC entities	0.96
7	A BENEFICIARY argument is a PER entity	0.93
8	A GEN-AFF relation exists between ORG and GPE entities	0.90

	Negative Feature	Weight
9	An entity has an ORG-AFF relation with multiple entities	-3.21
10	An entity has an PART-WHOLE relation with multiple entities	-2.49
11	An event has two PLACE arguments	-2.47
12	A TRANSPORT event has multiple DESTINATION arguments	-2.25
13	An entity has a GEN-AFF relation with multiple entities	-2.02
14	An ATTACK event has multiple PLACE arguments	-1.86
15	An entity has a PHYS relation with multiple entities	-1.69
16	An event has multiple VICTIM arguments	-1.61

Challenges

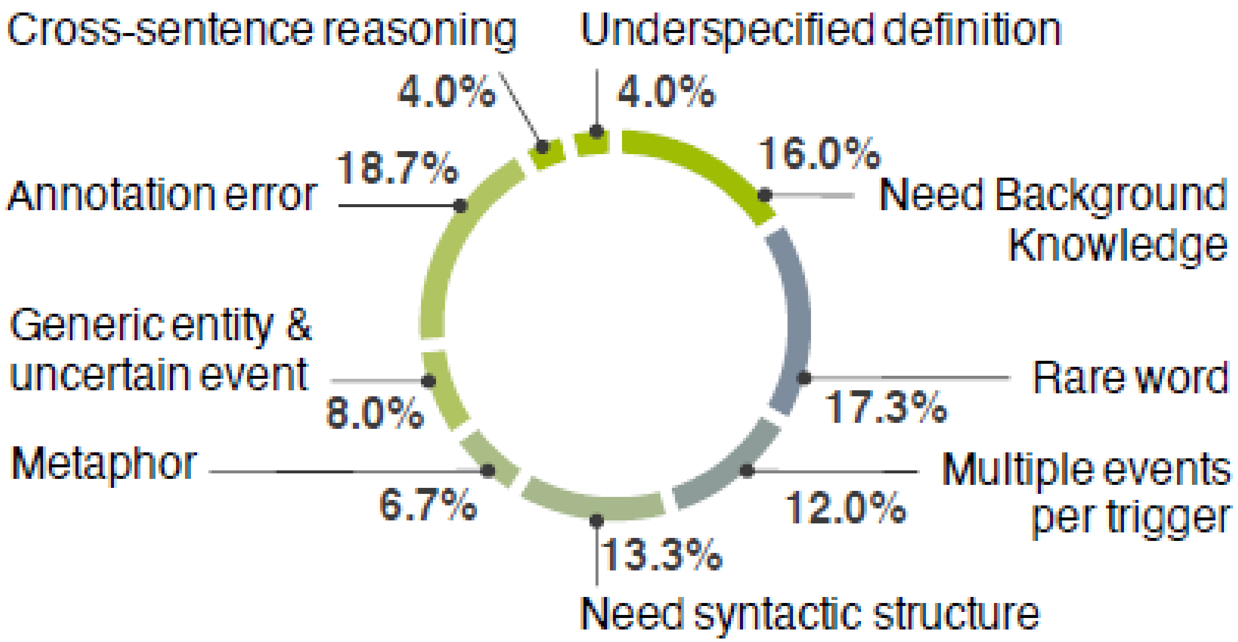


Figure 5: Distribution of remaining errors.

$$S_i = \frac{e^{V_i}}{\sum_j e^{V_j}}$$

Gold	[0,1,0]	
Pred(一次softmax)	[0.01,0.98,0.01]	loss:0.02
	$\downarrow$ Softmax	
Pred(两次softmax)	[0.2123, 0.5754, 0.2123]	loss:0.87