NLP Applications II

Event Extraction

Event Detection

- Goal: identify event triggers
- optional: classify them by type. E.g. *Life Movement, Transaction, Business, Conflict, Contact*

RELATION **directed**(m1, m2)

[The Big Sleep]$_{m1}$ is a 1946 film noir directed by [Howard Hawks]$_{m2}$, the first film version of Raymond Chandler's 1939 novel of the same name.

Event Detection

- Goal: identify event triggers
- optional: classify them by type. E.g. *Life Movement, Transaction, Business, Conflict, Contact*

EVENT **directed**

The Big Sleep is a 1946 film noir directed by Howard Hawks, the first film version of Raymond Chandler's 1939 novel of the same name.

Event Detection

- Event triggers can be verbs, nouns, adjectives, adverbs.

event	type
It **rained** last night.	verb
Her father is **retired.**	adjective
The **rioting** crowd approached the Capitol	modifier
The **attack** killed 7 and injured 20.	noun

More than trigger detection

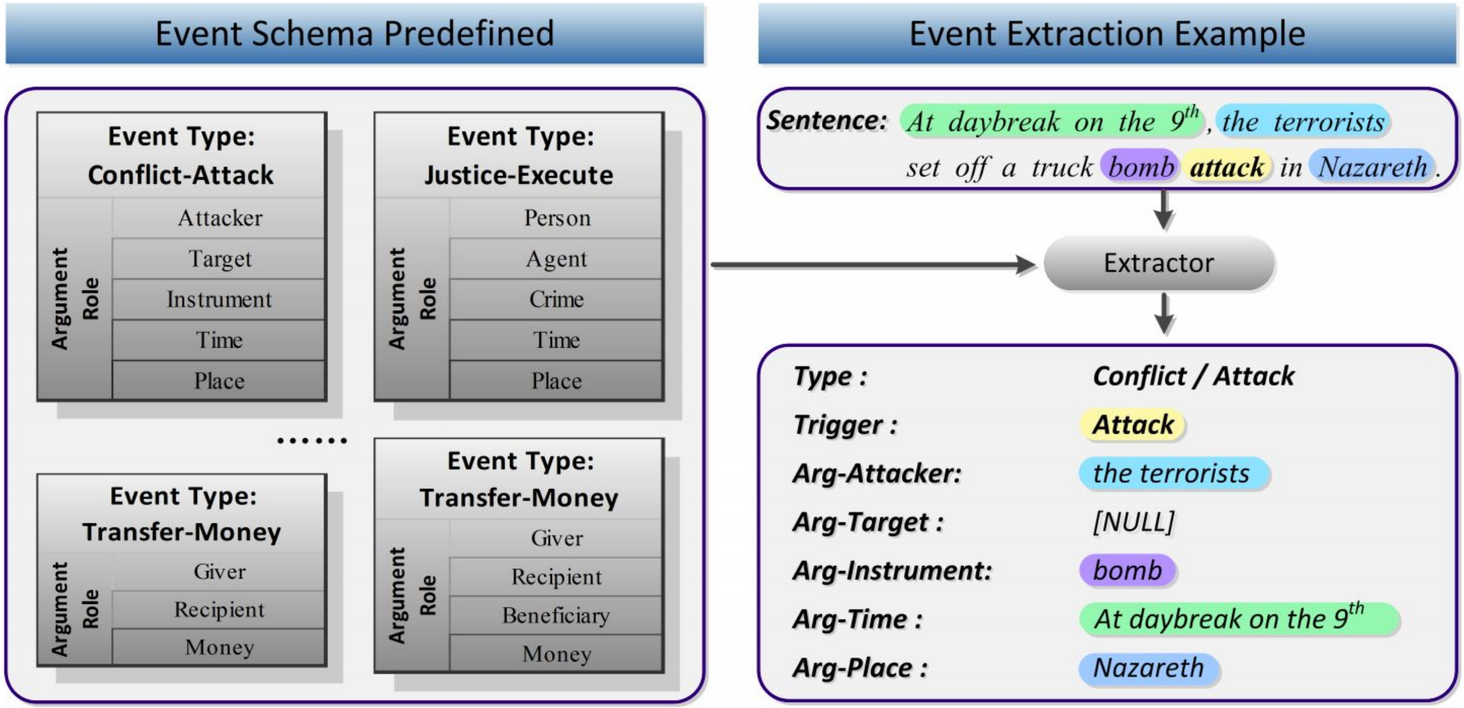
- Discovering event related information from text
- *Who did what, when, where, why and how*
- Event trigger identification + argument extraction
- Most of the time the goal is to map unstructured text to predefined event schema (closed event extraction)

Closed-domain event extraction

- **Event mention**: a phrase or sentence describing an event, including a trigger and several arguments.
- **Event trigger**: the main word that most clearly expresses an event occurrence, typically a verb or a noun.
- **Event argument**: an entity mention, temporal expression or value that serves as a participant or attribute with a specific role in an event.
- **Argument role**: the relationship between an argument to the event in which it participants

Note that same event can be mentioned in multiple event mentions (event co-reference)

Closed-domain event extraction



Source: Wei Xiang & Bang Wang, 2021

Open-domain event extraction

- No predefined event-schemas.
- Task consist in extracting events from text and clustering similar events.

Open-domain event extraction

Topic Detection and Tracking task:

Detection

- **Story segmentation**: detecting the boundaries of a story from news articles.
- **First story detection**: detecting the story that discuss a new topic in the stream of news.

clustering

- **Topic detection**: grouping the stories based on the topics they discuss.
- **Topic tracking**: detecting stories that discuss a previously known topic.
- **Story link detection**: deciding whether a pair of stories discuss the same topic.

Programs

- **Message Understanding Conference** - **MUC** (1987-1997): Extract slot-value structured from predefined schemas.
- **Automatic Content Extraction** - **ACE** (1999-2008): Entity and relation extraction (2000-2004), event extraction (2005)
- **Entities, Relation, Events** - **ERE** (2013-): Continuation of ACE program.
- **Knowledge Based Population** - **KBP** (2009-): Aims to extract information from large corpus to complete information missing in a knowledge base.
- **Topic Detection and Tracking** - **TDT** (1996-1997): Finding new events in a stream of broadcast news article
- TimeBank, BioNLP...

ACE 2005

SN	Event Type	SN	Event subtype
1	Life	1-5	Be-Born, Marry, Divorce, Injure, Die
2	Movement	6	Transport
3	Contact	7-8	Meet, Phone-write
4	Conflict	9-10	Attack, Demonstrate
5	Business	11-14	Merge-org, Declare-bankruptcy, Start-Org, End-org
6	Transaction	15-16	Transfer-money, Transfer-ownership
7	Persosnnel	17-20	Elect, Start-position, End-position, Nominate
8	Justice	21-33	Arrest-jail, Execute, Pardon, Release-parole, Fine, Convict, Charge-indict, Trial-hearing, Acquite, Sentence, Sue, Extradite, appeal

- English, Arabic and Chinese languages
- 599 annotated documents
- 6000 labeled events
- Newswire article, broadcast news...

TAC-KBP 2015

- The event types and subtypes from TACKBP (Rich ERE) are defined referring to the ACE corpus: 9 event types and 38 subtypes.
- 158 annotated documents (training) and 202 documents for testing
- Events must be assigned to three REALIS:
    - ACTUAL (actually occurred)
    - GENERIC (without specific time or place)
    - OTHERS (non-generic events, such as failed events, future events, and conditional statements etc.)

TAC-KBP 2015

Type	Subtype	Type	Subtype	Type	Subtype
Business	Start Org	Life	Divorce	Justice	Release-Parole
Business	End Org	Life	Injure	Justice	Trial-Hearing
Business	Declare Bankruptcy	Life	Die	Justice	Sentence
Business	Merge Org	Transaction	Transfer Ownership	Justice	Fine
Conflict	Attack	Transaction	Transfer Money	Justice	Charge-Indict
Conflict	Demonstrate	Transaction	Transaction	Justice	Sue
Contact	Meet	Movement	Transport.Person	Justice	Extradite
Contact	Correspondence	Movement	Transport.Artifact	Justice	Acquit
Contact	Broadcast	Personnel	Start Position	Justice	Convict
Contact	Contact	Personnel	End Position	Justice	Appeal
Manufacture	Artifact	Personnel	Nominate	Justice	Execute
Life	Be Born	Personnel	Elect	Justice	Pardon
Life	Marry	Justice	Arrest-Jail		

Table 1: Event Types and Subtypes

GENIA

Event Type	Primary Argument	Secondary Argument
Gene_expression	Theme(Protein)	
Transcription	Theme(Protein)	
Localization	Theme(Protein)	Loc(Entity)?
Protein_catabolism	Theme(Protein)	
Binding	Theme(Protein)+	Site(Entity)*
Protein_modification	Theme(Protein), Cause(Protein/Event)?	Site(Entity)?
Phosphorylation	Theme(Protein), Cause(Protein/Event)?	Site(Entity)?
Ubiquitination	Theme(Protein), Cause(Protein/Event)?	Site(Entity)?
Acetylation	Theme(Protein), Cause(Protein/Event)?	Site(Entity)?
Deacetylation	Theme(Protein), Cause(Protein/Event)?	Site(Entity)?
Regulation	Theme(Protein/Event), Cause(Protein/Event)?	Site(Entity)?, CSite(Entity)?
Positive_regulation	Theme(Protein/Event), Cause(Protein/Event)?	Site(Entity)?, CSite(Entity)?
Negative_regulation	Theme(Protein/Event), Cause(Protein/Event)?	Site(Entity)?, CSite(Entity)?

Table 1: Event types and their arguments for Genia Event Extraction task. The type of each filler entity is specified in parenthesis. Arguments that may be filled more than once per event are marked with "+", and optional arguments are with "?".

Evaluation

ACE 2005 dataset with the standard evaluation procedures as follows:

- **Trigger detection**: A trigger is correctly detected if its offsets (viz., the position of the trigger word in text) match a reference trigger.
- **Type identification**: An event type is correctly identified if both the trigger's offset and event type match a reference trigger and its event type.
- **Argument detection**: An argument is correctly detected if its offsets match any of the reference argument mentions (viz., correctly recognizing participants in an event).
- **Role identification**: An argument role is correctly identified if its event type, offsets, and role match any of the reference argument mentions.

Precision, Recall and F1-score over predicted vs gold-annotations

Approaches

- Traditional feature based ML vs Deep Learning
- Pipeline classification models vs Joint models
- Prompt based

Pipeline classification

Two stages:

1. Trigger detection and event type identification
2. Given the trigger argument detection (entities, time and values) and argument role identification

Cross-lingual XLM-R pipeline

Leverage pre-trained LM, train in English and test in Arabic.

Pipeline with two fine-tuned XLM-R modules in sequence:

1) **Detect events**: fine-tune XLM-R for token classification using BIO tags

*Local news outlets **reported** that two Kazakhs and a Chechen were **killed**.*

Cross-lingual XLM-R pipeline

Leverage pre-trained LM, train in English and test in Arabic.

Pipeline with two fine-tuned XLM-R modules in sequence:

1) **Detect events**: fine-tune XLM-R for token classification using BIO tags

event | event

Local news outlets ***reported*** *that two Kazakhs and a Chechen were* ***killed****.*

B-Verbal#Neutral | B-Material#Harmful

Cross-lingual XLM-R pipeline

Leverage pre-trained LM, train in English and test in Arabic.

Pipeline with two fine-tuned XLM-R modules in sequence:

2) **Detect arguments**: <u>For each event</u>, fine-tune XLM-R model for token classification using BIO tags and $$$ special token

Cross-lingual XLM-R pipeline

Leverage pre-trained LM, train in English and test in Arabic.

Pipeline with two fine-tuned XLM-R modules in sequence:

2) **Detect arguments**: <u>For each event</u>, fine-tune XLM-R model for token classification using BIO tags and $$$ special token

event

Local news outlets ***$$$*** *reported* ***$$$*** *that two Kazakhs and a Chechen were killed.*

event

Local news outlets reported that two Kazakhs and a Chechen were ***$$$*** *killed* ***$$$****.*

Cross-lingual XLM-R pipeline

Leverage pre-trained LM, train in English and test in Arabic.

Pipeline with two fine-tuned XLM-R modules in sequence:

2) **Detect arguments**: For each event, fine-tune XLM-R model for token classification using BIO tags and $$$ special token

agent event patient

Local news outlets ***$$$*** *reported* ***$$$*** *that two Kazakhs and a Chechen were killed.*

B-agent I-agent I-agent B-patient

event

Local news outlets reported that two Kazakhs and a Chechen were ***$$$*** *killed* ***$$$****.*

Crosslingual XLM-R pipeline

Leverage pre-trained LM, train in English and test in Arabic.

Pipeline with two fine-tuned XLM-R modules in sequence:

2) **Detect arguments**: For each event, fine-tune XLM-R model for token classification using BIO tags and $$$ special token

agent event patient

Local news outlets ***$$$*** *reported* ***$$$*** *that two Kazakhs and a Chechen were killed.*

B-agent I-agent I-agent B-patient

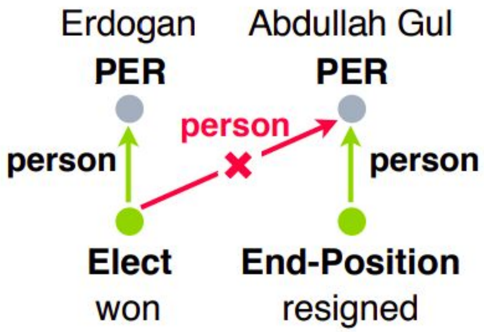
patient patient event

Local news outlets reported that two Kazakhs and a Chechen were ***$$$*** *killed* ***$$$****.*

B-patient I-patient B-patient I-patient

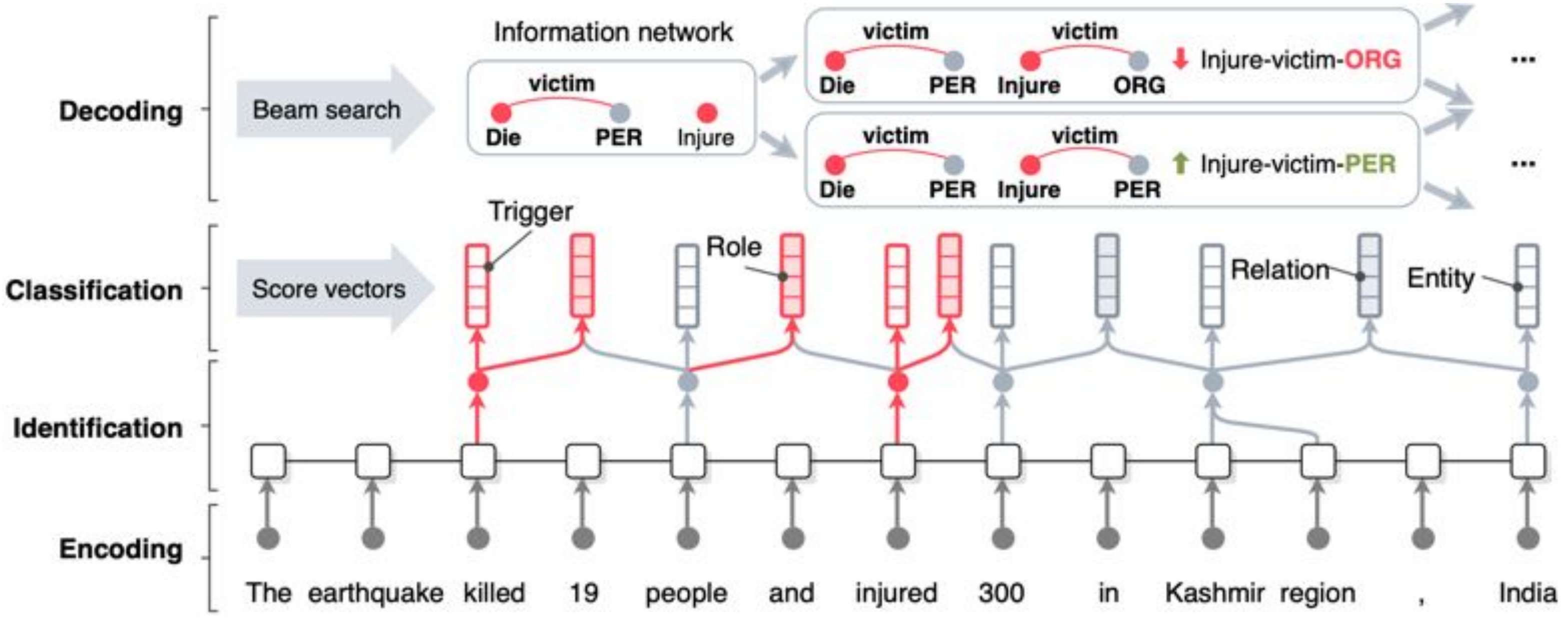
Joint Event Extraction

- Pipeline model suffer from the *error propagation* problem
- In pipeline models inter-dependencies of different subtasks cannot well exploited
- Joint models extract triggers and corresponding arguments simultaneously.



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

OneIE: An End-to-end Neural Model for IE

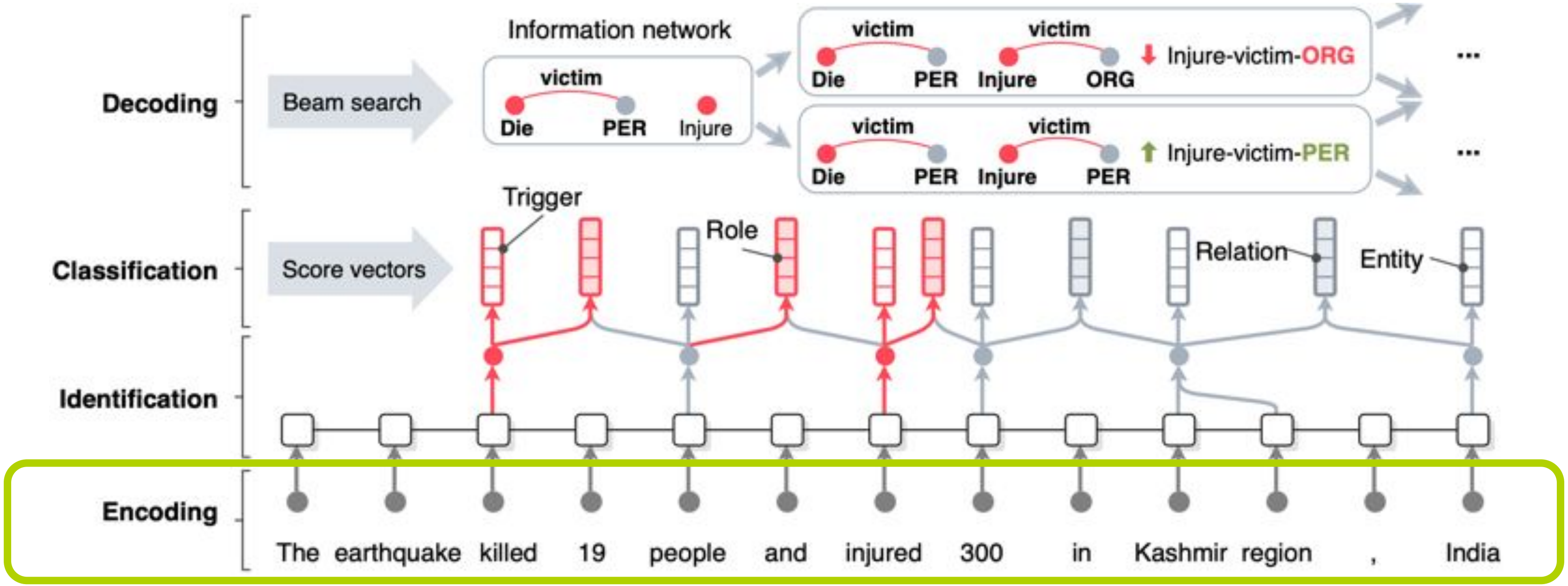


- Our OneIE framework extracts the information graph from a given sentence in four steps: encoding, identification, classification, and decoding

Slice source: https://blender.cs.illinois.edu/paper/jointiesentence2020.pptx

Lin e al., 2020

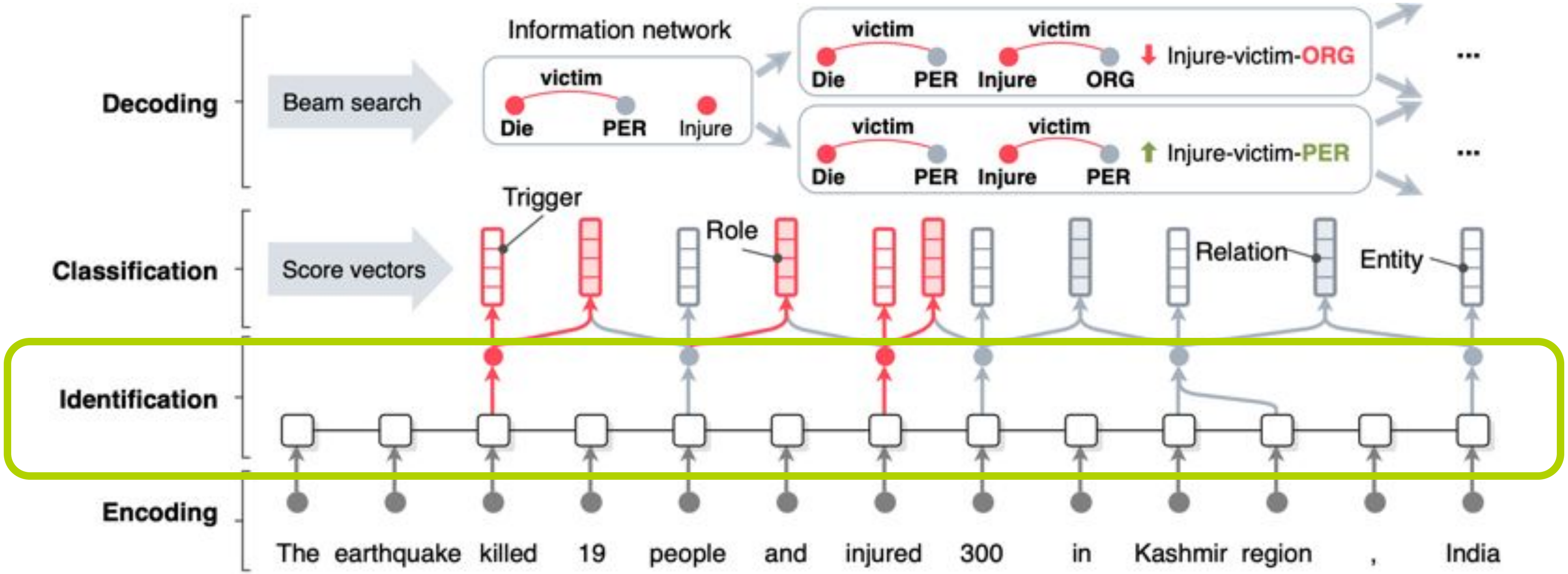
OneIE: An End-to-end Neural Model for IE



- **Encoding**: We use a BERT encoder to obtain the contextualized embedding of each token

Lin et al., 2020

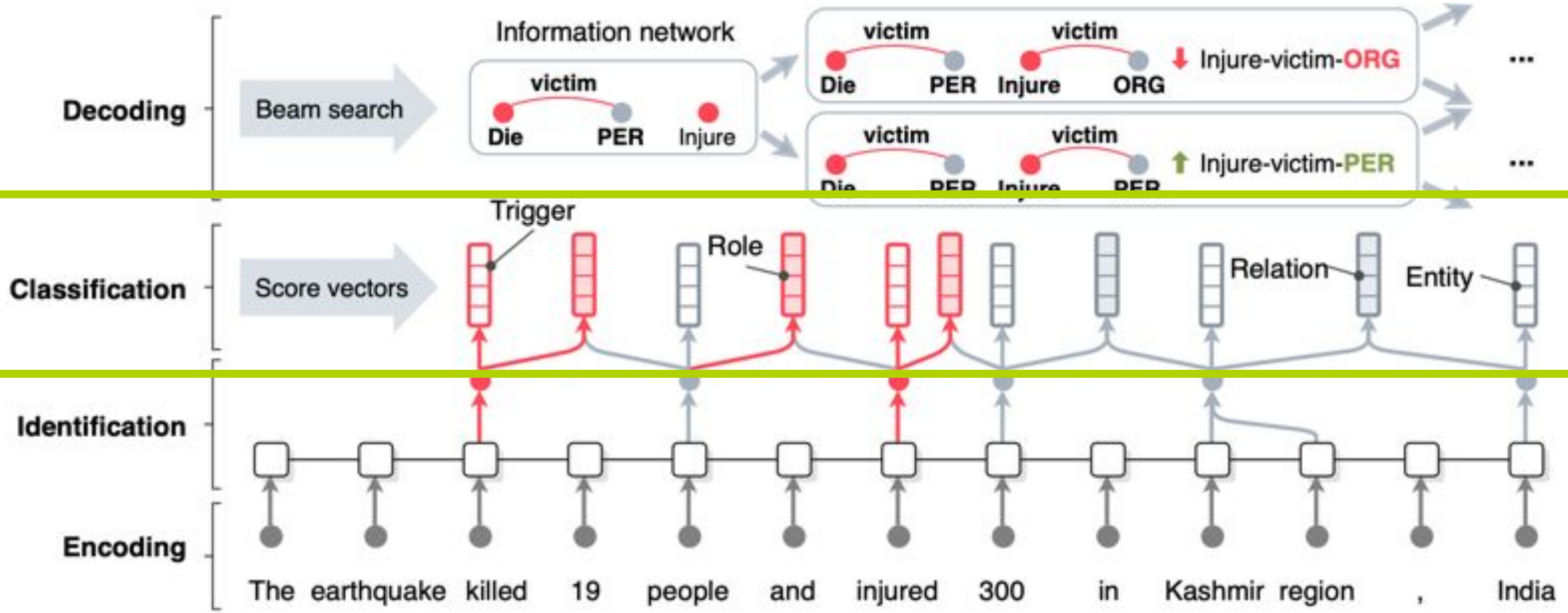
OneIE: An End-to-end Neural Model for IE



- **Identification**: We use CRF taggers to identify entity mentions and event triggers
- We define the identification loss as $\mathcal{L}^{\mathrm{I}} = -\log p(\boldsymbol{z}|\boldsymbol{X})$

Lin et al., 2020

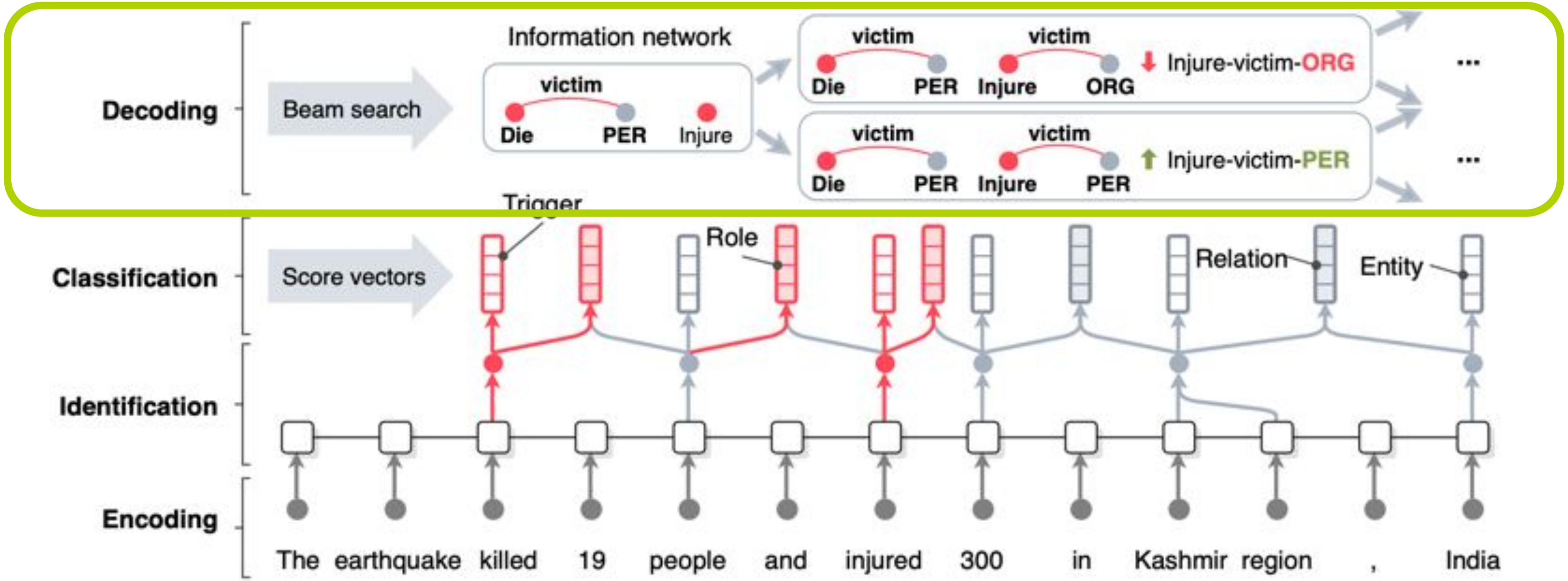
OneIE: An End-to-end Neural Model for IE



- **Classification**: We use task-specific feed-forward networks to calculate label scores for each node or edge
- We define the classification loss as $\mathcal{L}^{t} = -\frac{1}{N^t}\sum_{i=1}^{N^t} \boldsymbol{y}_i^t \log \hat{\boldsymbol{y}}_i^t$

Lin et al., 2020

OneIE: An End-to-end Neural Model for IE

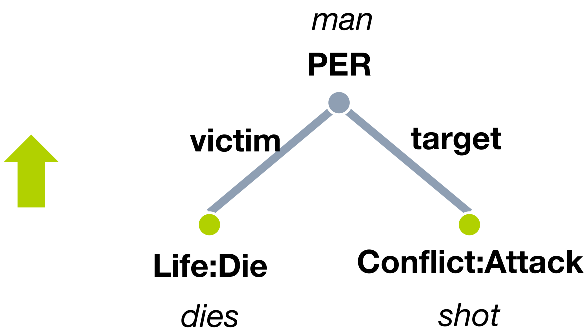


- **Decoding**: In the test phase, we use a beam search decoder to find the information graph with the highest global score

Lin et al., 2020

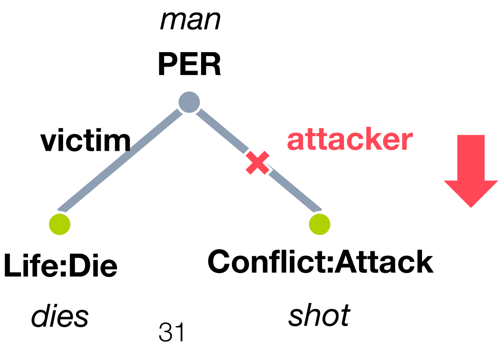
Incorporating Global Features

- We design a set of *global feature templates* (e.g., *event_type*$_1$ *– role*$_1$ *– role*$_2$ *–* *event_type*$_2$: an entity acts a role$_1$ argument for an event_type$_1$ event and a role$_2$ argument for an event_type$_2$ event in the same sentence)
- The model learns the *weight* of each feature during training



Life:Die – victim – target – Conflict:Attack

Positive weight



Life:Die – victim – attacker – Conflict:Attack

Negative weight

Lin et al., 2020

Incorporating Global Features

- Given a graph G, we generate its global feature vector as $\boldsymbol{f}_G = \{f_{1(G)}, \ldots, f_M(G)\}$, where $f_i(\cdot)$ 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, we learn a weight vector $\boldsymbol{u} \in \mathbb{R}^M$ and calculate the global feature score of G as the dot production of $\boldsymbol{f}_G$ and $\boldsymbol{u}$.
- **Global score** of a graph: local graph score + global feature score:

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

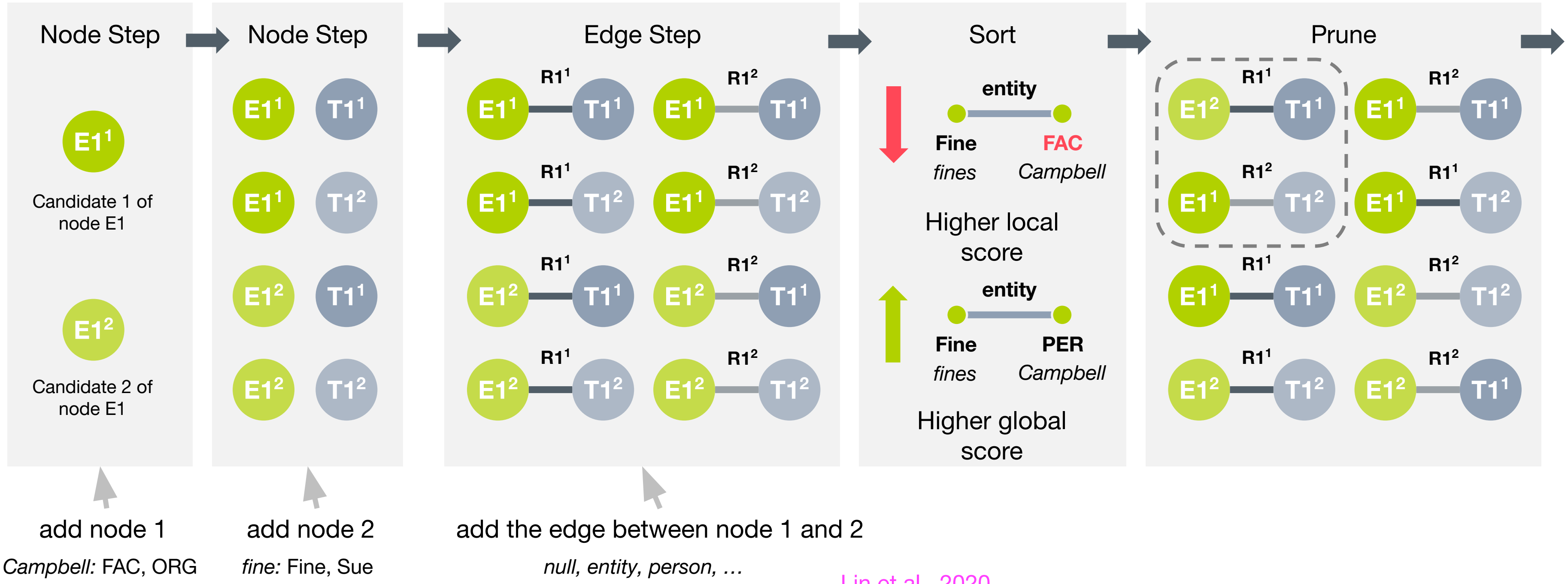
- We assume that the gold-standard graph for a sentence should achieve the highest global score and minimize the following loss function:

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

Lin et al., 2020

Decoding

- We use beam search to decode the information graph
- Example: *He also brought a check from **Campbell** to pay the **fines** and fees.*



Lin et al., 2020

Event Coreference

HP acquires Electronic Data Systems

Document 1

Hewlett-Packard is negotiating to buy technology services provider Electronic Data Systems.

With a market value of about $115 billion, HP could easily use its own stock to finance the purchase

If the deal is completed, it would be HP's biggest acquisition since it bought Compaq Computer Corp. for $19 billion in 2002.

Document 2

Industry sources have confirmed to eWEEK that Hewlett Packard will acquire ElectronicData Systems for about $13 billion

Bejan and Harabagiu, 2014

Event Coreference

- Similar to named entity coreference, we can train a binary classifier to predict the probability of coreference for each pair of event mentions.
- Features include similarity measures between event triggers and event arguments:
- Evaluation: B^3 Precision and Recall

Factuality Detection

I locked the door

I think I locked the door

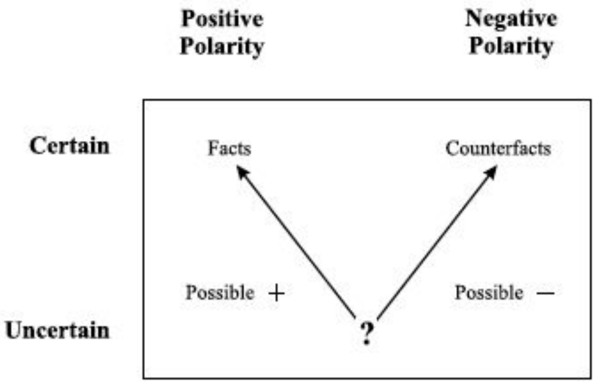


Figure 1
The double range of factuality.

I forgot to lock the door

I possibly forgot to lock the door

Sauri et al., 2012

Factuality Detection

(1) U.S. embassies and military installations around the world were ordered(3.0) to set(2.6) up barriers and tighten(2.6) security to prevent(1.8) easy access(-2.4) by unauthorized people.

(2) Intel's most powerful computer chip has flaws that could delay(0.8) several computer makers' marketing efforts(2.6), but the "bugs" aren't expected(-2.6) to hurt(-2.0) Intel.

(3) President Bush on Tuesday said(3.0) the United States may extend(1.6) its naval quarantine(2.6) to Jordan's Red Sea port of Aqaba to shut(1.4) off Iraq's last unhindered trade route.

(4) He also said(3.0) of trade(-0.8) with Iraq: "There are no shipments at the moment."

Lee et al., 2015

Factuality Features

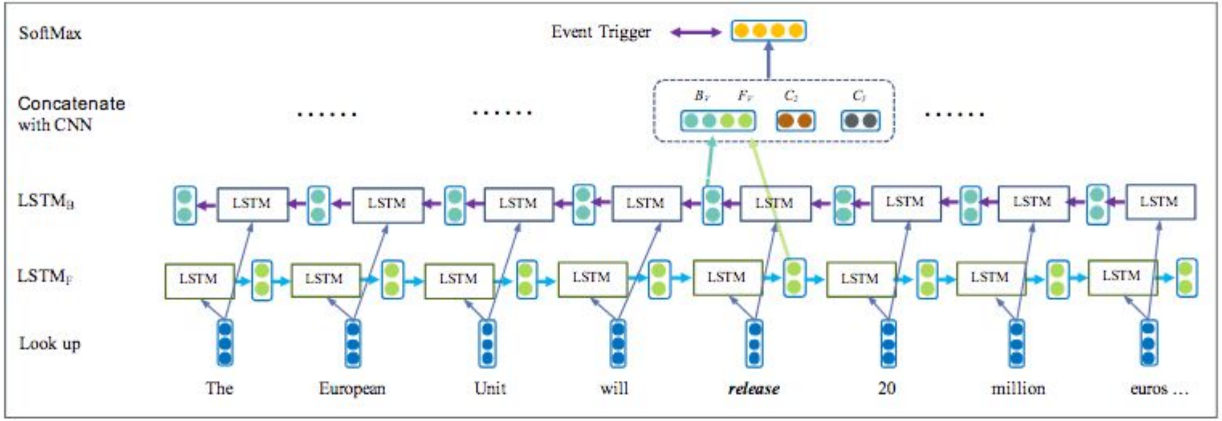
Score = features(x)^T * W

feature
token lemma, part of speech, dependency
token lemma, part of speech, dependency
presence hedge words (probably, possibly, maybe, …)
presence of implicative verbs (manage to vs. forget to)

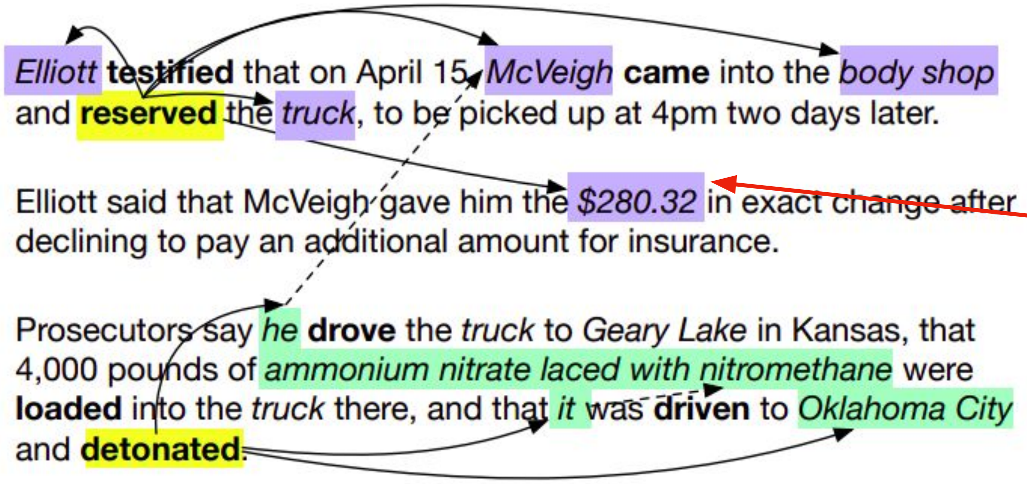
Neural Factuality Detection

- Same architecture as event detection models, but predict a scalar value rather than a binary event indicator

+3



Document level Event Extraction



Document level
argument extraction

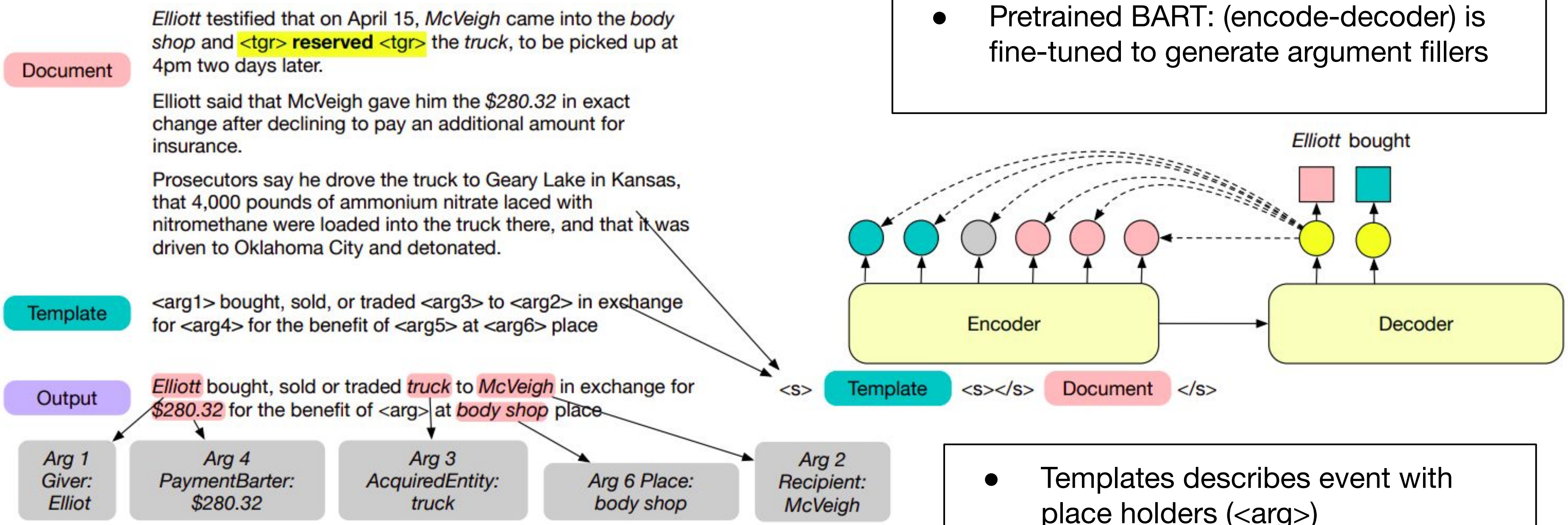
Argument Transaction.Exchange for reserved can only found in the next sentence.

This work purposes de use of conditioned text generation for document level argument extraction, without NER and Event Coreference.

Li, Ji and Han 2021

Document level Argument Extraction

- Pretrained BART: (encode-decoder) is fine-tuned to generate argument fillers



- Templates describes event with place holders (<arg>)
- One template per event type extracted from ontology
- Triggers *event markers*

Li, Ji and Han 2021

Document level Argument Extraction

- Generation probability of argument token is computed by the dot product of the decoder output and the embeddings in the input documents.

$$p(x_i = w | x_{<i}, c, t) = \begin{cases} \text{Softmax}\left(h_i^T \text{Emb}(w)\right) & w \in V_c \\ 0 & w \notin V_c \end{cases}$$

Prevents model from hallucinating arguments

- The model is trained by minimizing the negative log-likehood over all content, template, outputs in the dataset D:

$$\mathcal{L}(D) = -\sum_{i=1}^{|D|} \log p_\theta \left(x^i \mid c^i\right)$$

Li, Ji and Han 2021

Label Verbalization and Entailment for Effective Zero- and Few-Shot Relation Extraction

Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena and Eneko Agirre

EMNLP 2021

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea

HiTZ
Hizkuntza Teknologiako Zentroa
Basque Center for Language Technology

Relation Extraction task

Given 2 entities $e1$ and $e2$ and a context c, predict the semantic relation (if any) holding between the two entities in the context.

⟨**Billy Mays**$_{\text{PERSON}}$, **Tampa**$_{\text{CITY}}$⟩

Billy Mays, the bearded, boisterous pitchman who, as the undisputed king of TV yell and sell, became an unlikely pop culture icon, died at his home in **Tampa**, Fla, on Sunday.

→ `per:city_of_death`

Current state of the art

- Mostly approached via **supervised learning** on large datasets or via **distant-supervision** when a Knowledge Base and a large set of documents are available.

- Supervised learning:
  - Pretrained language models (LM) fine-tuned on large amount of labeled data.
  - Focused on **models**: finding better pre-training objectives, relation representations or incorporating external knowledge.

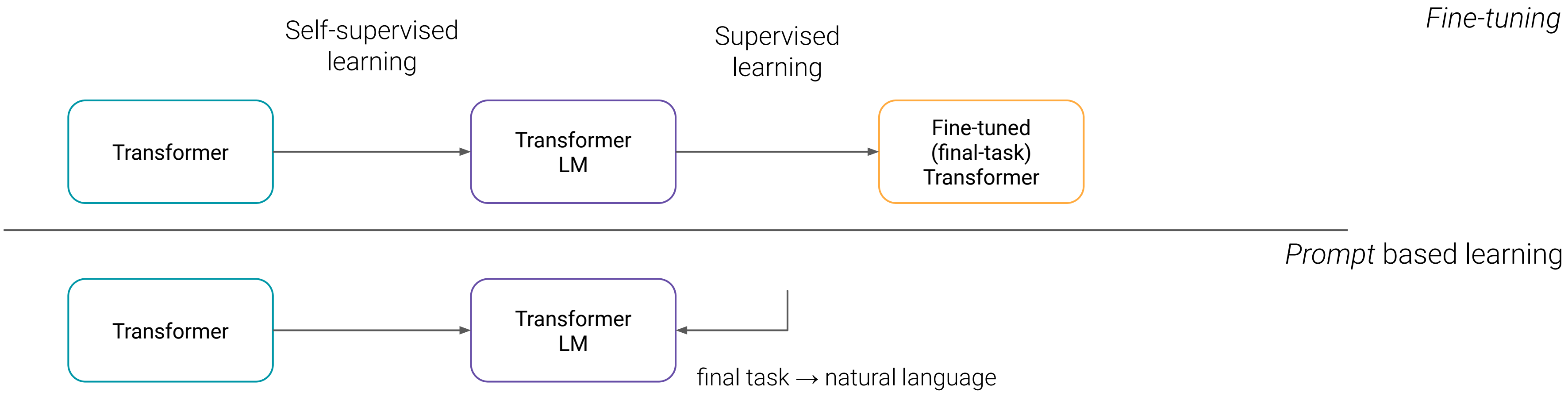
- Distant-supervision:
  - Pretrained language models (LM) fine-tuned on noisy large amounts of labeled data.
  - Focused on **data**: alleviating the noisy signal from the data and finding better bag of context representations.

- How about focusing on <u>a model that works with a small amount of data?</u>

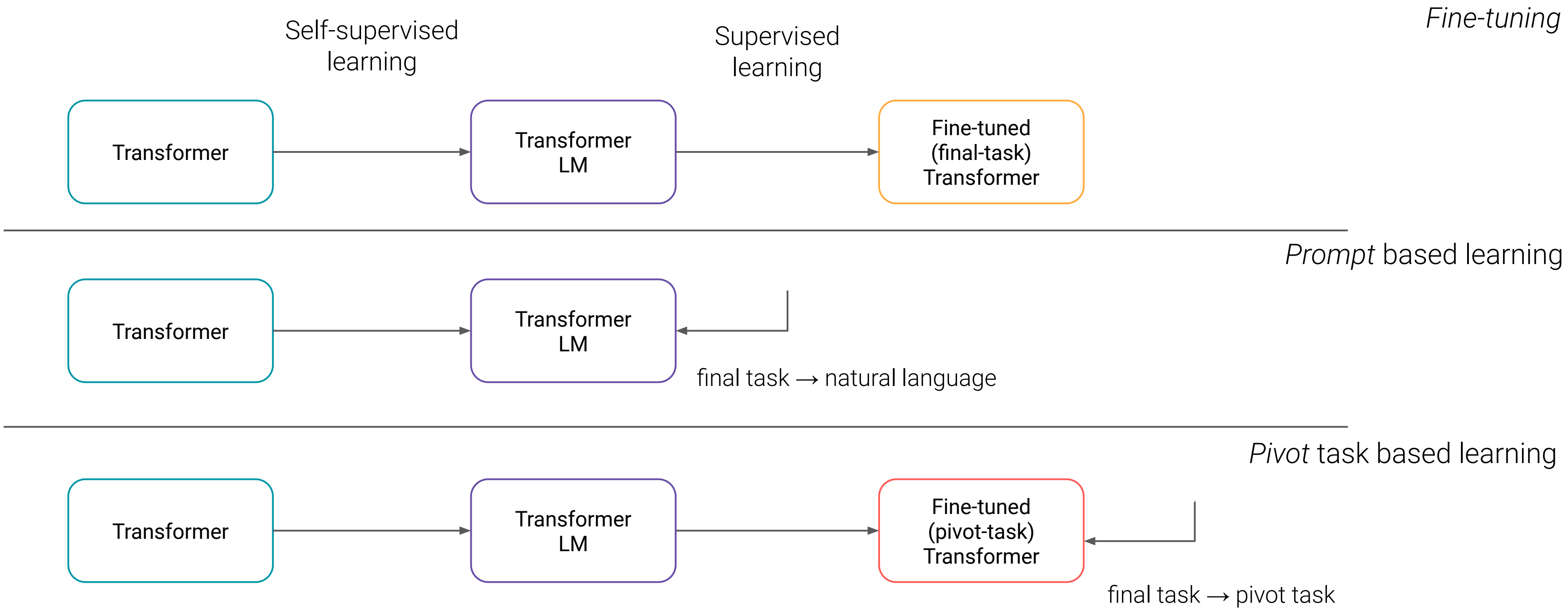
Alternative paradigms to fine-tuning

Fine-tuning

Alternative paradigms to fine-tuning



Alternative paradigms to standard fine-tuning



Entailment-based Relation Extraction

- We propose to reformulate Relation Extraction as a Textual Entailment (similar to Obamuyide and Vlachos (2018)[1]).

premise: hypotesis:

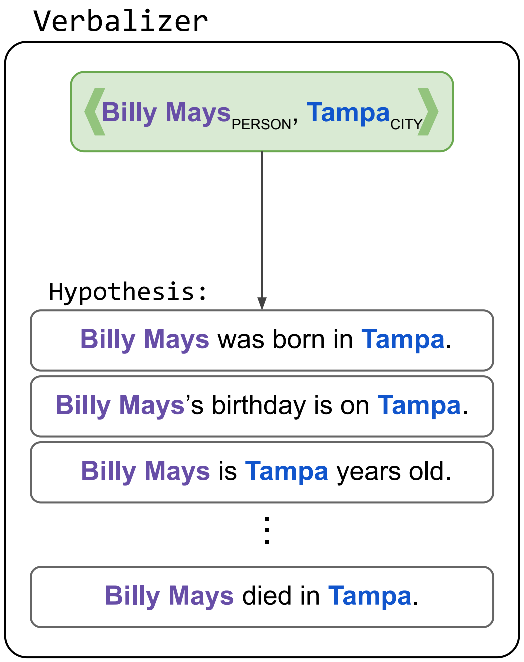
Context | **Billy Mays** died in **Tampa**. → E → per:city_of_death

- We found **T**extual **E**ntailment (aka **NLI**) to be a robust **pivot task** for zero- and few-shot learning.
- We propose a simple yet effective **inference strategy** based on NLI pretrained models to achieve competent results even with no training examples.

[1]Abiola Obamuyide and Andreas Vlachos. 2018. Zero-shot relation classification as textual entailment. In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pages 72–78, Brussels, Belgium. Association for Computational Linguistics.

Approach

Entailment based Relation Extraction



- Function that combines entity pairs with **templates** to generate textual hypotheses for relations:

$$hyp = \text{VERBALIZE}(t, x_{e1}, x_{e2})$$

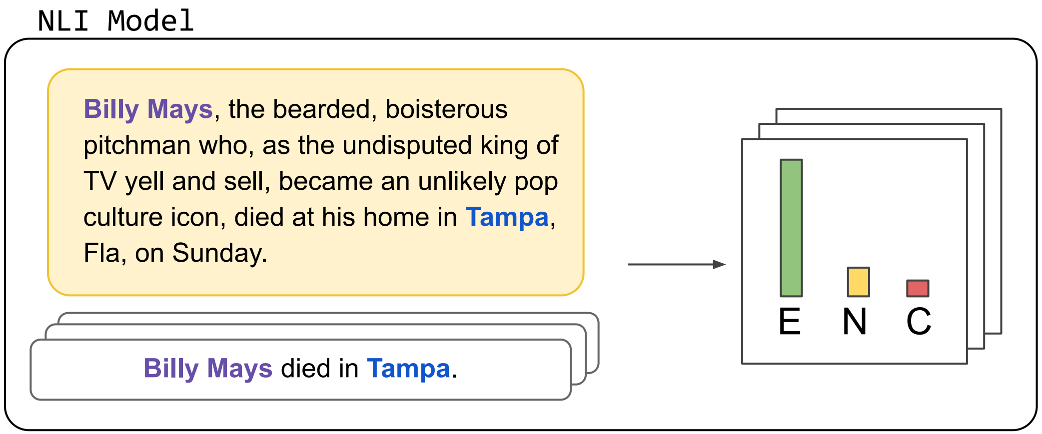
- N:M relation between templates and relations
- Also, type constraints for entities

Relation	Templates	Valid argument types
per:alternate_names	{subj} is also known as {obj}	PERSON, MISC
per:date_of_birth	{subj}'s birthday is on {obj}	DATE
	{subj} was born on {obj}	
per:age	{subj} is {obj} years old	NUMBER, DURATION
per:country_of_birth	{subj} was born in {obj}	COUNTRY
per:stateorprovince_of_birth	{subj} was born in {obj}	STATE_OR_PROVINCE
per:city_of_birth	{subj} was born in {obj}	CITY, LOCATION

Entailment based Relation Extraction

- Next, we compute the entailment probabilities for each of the hypothesis independently.

$$\mathrm{P}_{NLI}(x, hyp)$$



Entailment based Relation Extraction

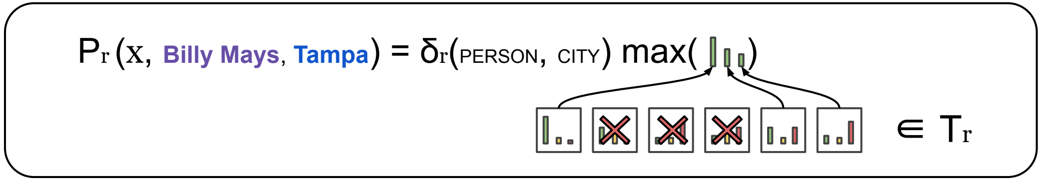
- We compute the probability of relation r based on the hypothesis probabilities and entity constraints:

$$\mathsf{P}_r(x, x_{e1}, x_{e2}) = \delta_r(e_1, e_2) \max_{t \in T_r} \mathsf{P}_{NLI}(x, hyp)$$

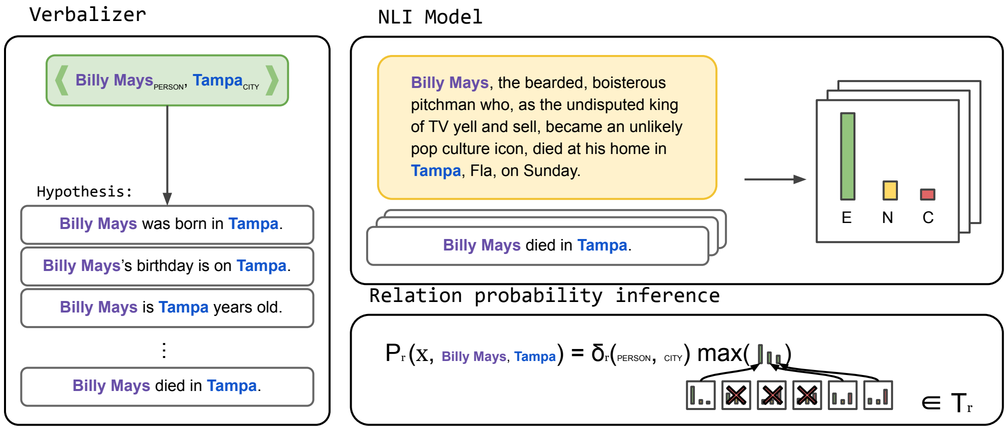
- The δ_r function describes the entity constraints of the relation r:

$$\delta_r(e_1, e_2) = \begin{cases} 1 & e_1 \in E_{r1} \wedge e_2 \in E_{r2} \\ 0 & \text{otherwise} \end{cases}$$

Relation probability inference



Entailment based Relation Extraction



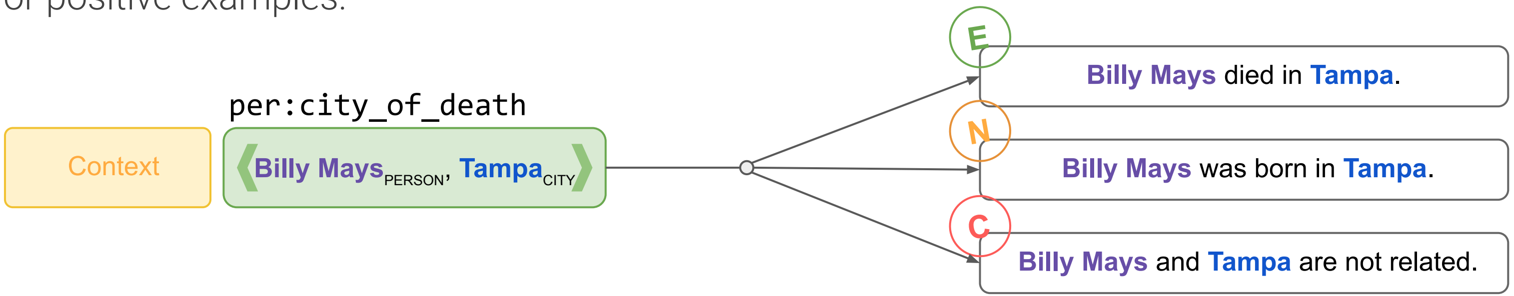
Finally, we return the relation with the higher probability:

$$\hat{r} = \arg\max_{r \in R} \mathrm{P}_r(x, x_{e1}, x_{e2})$$

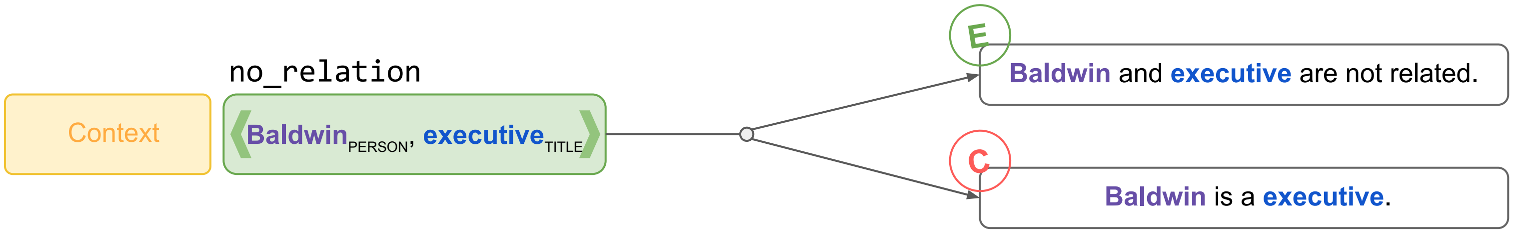
If no relation has higher entailment prob. than threshold (T, default 0.5), r = `no_relation`.

Fine-tuning NLI on Relation Extraction data

For positive examples:



For negative examples:



Results

Zero-Shot

NLI Model	# Param.	MNLI Acc.
ALBERT$_{xxLarge}$	223M	90.8
RoBERTa	355M	90.2
BART	406M	89.9
DeBERTa$_{xLarge}$	900M	91.7
DeBERTa$_{xxLarge}$	1.5B	91.7

Zero-Shot relation extraction:

Zero-Shot

NLI Model	# Param.	MNLI Acc.	No Dev ($\mathcal{T} = 0.5$) Pr.	Rec.	F1
$\text{ALBERT}_{\text{xxLarge}}$	223M	90.8	32.6	**79.5**	46.2
RoBERTa	355M	90.2	32.8	75.5	45.7
BART	406M	89.9	39.0	63.1	48.2
$\text{DeBERTa}_{\text{xLarge}}$	900M	91.7	40.3	77.7	53.0
$\text{DeBERTa}_{\text{xxLarge}}$	1.5B	91.7	**46.6**	76.1	**57.8**

Zero-Shot relation extraction:

- Default threshold for `no_relation` produces low precision

Zero-Shot

NLI Model	# Param.	MNLI Acc.	No Dev ($\mathcal{T} = 0.5$) Pr.	Rec.	F1	1% Dev Pr.	Rec.	F1
$\text{ALBERT}_{\text{xxLarge}}$	223M	90.8	32.6	**79.5**	46.2	55.2	58.1	56.6 $\pm$1.4
RoBERTa	355M	90.2	32.8	75.5	45.7	58.5	53.1	55.6 $\pm$1.3
BART	406M	89.9	39.0	63.1	48.2	60.7	46.0	52.3 $\pm$1.8
$\text{DeBERTa}_{\text{xLarge}}$	900M	91.7	40.3	77.7	53.0	**66.3**	59.7	**62.8** $\pm$1.7
$\text{DeBERTa}_{\text{xxLarge}}$	1.5B	91.7	**46.6**	76.1	**57.8**	63.2	**59.8**	61.4 $\pm$1.0

Zero-Shot relation extraction:

- Default threshold for `no_relation` produces low precision
- With 1% of Dev (2 examples per relation, 100 negative examples) threshold can be tuned for each relation, yielding better results

Zero-Shot

NLI Model	# Param.	MNLI Acc.	No Dev ($\mathcal{T} = 0.5$) Pr.	Rec.	F1	1% Dev Pr.	Rec.	F1
$\text{ALBERT}_{\text{xxLarge}}$	223M	90.8	32.6	**79.5**	46.2	55.2	58.1	56.6 $\pm$1.4
RoBERTa	355M	90.2	32.8	75.5	45.7	58.5	53.1	55.6 $\pm$1.3
BART	406M	89.9	39.0	63.1	48.2	60.7	46.0	52.3 $\pm$1.8
$\text{DeBERTa}_{\text{xLarge}}$	900M	91.7	40.3	77.7	53.0	**66.3**	59.7	**62.8** $\pm$1.7
$\text{DeBERTa}_{\text{xxLarge}}$	1.5B	91.7	**46.6**	76.1	**57.8**	63.2	**59.8**	61.4 $\pm$1.0

Zero-Shot relation extraction:

- Default threshold for `no_relation` produces low precision
- With 1% of Dev (2 examples per relation, 100 negative examples) threshold can be tuned for each relation, yielding better results
- DeBERTa achieves the best results, maybe due to the number of parameters.
- Note that minor variations in MNLI (±2) produce large variations in F1.

Few-Shot

Model	1%			5%			10%		
	Pr.	Rec.	F1	Pr.	Rec.	F1	Prec.	Rec.	F1
SpanBERT	0.0	0.0	0.0 ±0.0	36.3	23.9	28.8 ±13.5	3.2	1.1	1.6 ±20.7
RoBERTa	56.8	4.1	7.7 ±3.6	52.8	34.6	41.8 ±3.3	61.0	50.3	55.1 ±0.8
K-Adapter	73.8	7.6	13.8 ±3.4	56.4	37.6	45.1 ±0.1	62.3	50.9	56.0 ±1.3
LUKE	61.5	9.9	17.0 ±5.9	57.1	47.0	51.6 ±0.4	60.6	60.6	60.6 ±0.4

Few-Shot relation extraction:

- State of the art systems have difficulties to learn the task where very small amount of data is annotated. Indeed, SpanBERT does not even work.

Few-Shot

Model	1%			5%			10%		
	Pr.	Rec.	F1	Pr.	Rec.	F1	Prec.	Rec.	F1
SpanBERT	0.0	0.0	0.0 ±0.0	36.3	23.9	28.8 ±13.5	3.2	1.1	1.6 ±20.7
RoBERTa	56.8	4.1	7.7 ±3.6	52.8	34.6	41.8 ±3.3	61.0	50.3	55.1 ±0.8
K-Adapter	73.8	7.6	13.8 ±3.4	56.4	37.6	45.1 ±0.1	62.3	50.9	56.0 ±1.3
LUKE	61.5	9.9	17.0 ±5.9	57.1	47.0	51.6 ±0.4	60.6	60.6	60.6 ±0.4
$NLI_{RoBERTa}$ (ours)	56.6	55.6	56.1 ±0.0	60.4	68.3	64.1 ±0.2	**65.8**	69.9	67.8 ±0.2
$NLI_{DeBERTa}$ (ours)	**59.5**	**68.5**	**63.7** ±0.0	**64.1**	**74.8**	**69.0** ±0.2	62.4	**74.4**	**67.9** ±0.5

Few-Shot relation extraction:

- State of the art systems have difficulties to learn the task where very small amount of data is annotated. Indeed, SpanBERT does not even work.
- NLI systems instead, achieve very good results from the beginning, and, as the rest do, the results improve with training data.
- As in the zero-shot setting, DeBERTa model score the best.

Full training

Full trained relation extraction

- NLI sistems perform in pair when large amount of annotated data is available (RoBERTa vs $NLI_{RoBERTa}$).
- The performance gap between NLI systems is maintained even after fine-tuned with the whole dataset ($NLI_{RoBERTa}$ vs $NLI_{DeBERTa}$).
- We outperformed the state of the art with $NLI_{DeBERTa}$. But it is true that similar performance is expected using a vanilla DeBERTa trained on whole TACRED (Zhang et al. 2017)[3].

Model	Pr.	Rec.	F1
SpanBERT	70.8	70.9	70.8
RoBERTa	70.2	72.4	71.3
K-Adapter	70.1	74.0	72.0
LUKE	70.4	75.1	72.7
$NLI_{RoBERTa}$ (ours)	71.6	70.4	71.0
$NLI_{DeBERTa}$ (ours)	**72.5**	**75.3**	**73.9**

[3]Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017), pages 35–45.

Conclusions and Future Work

Conclusions and Future Work

Conclusions:

- Prompt or pivot-task based methods are effective for zero- or few-shot relation-extraction.
- Specially NLI based relation extraction yields very good results.
- NLI based methods are robust discriminating positive relations, but have difficulties deciding when a relation exists or not. Better methods for `no_relation` identification need to be developed.

Future Work:

- Develop a better method for `no_relation` (negative class) identification.
- Extend the framework to other tasks such as Event Argument Extraction.

Label Verbalization and Entailment for Effective Zero- and Few-Shot Relation Extraction

Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena and Eneko Agirre

EMNLP 2021

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea

HiTZ
Hizkuntza Teknologiako Zentroa
Basque Center for Language Technology