Relation extraction for non-standard types

Benjamin Roth, Ludwig Maximilian University Munich




work with Costanza Conforti, Nina Poerner, Sanjeev Karn, Hinrich Schütze

From unstructured to structured data

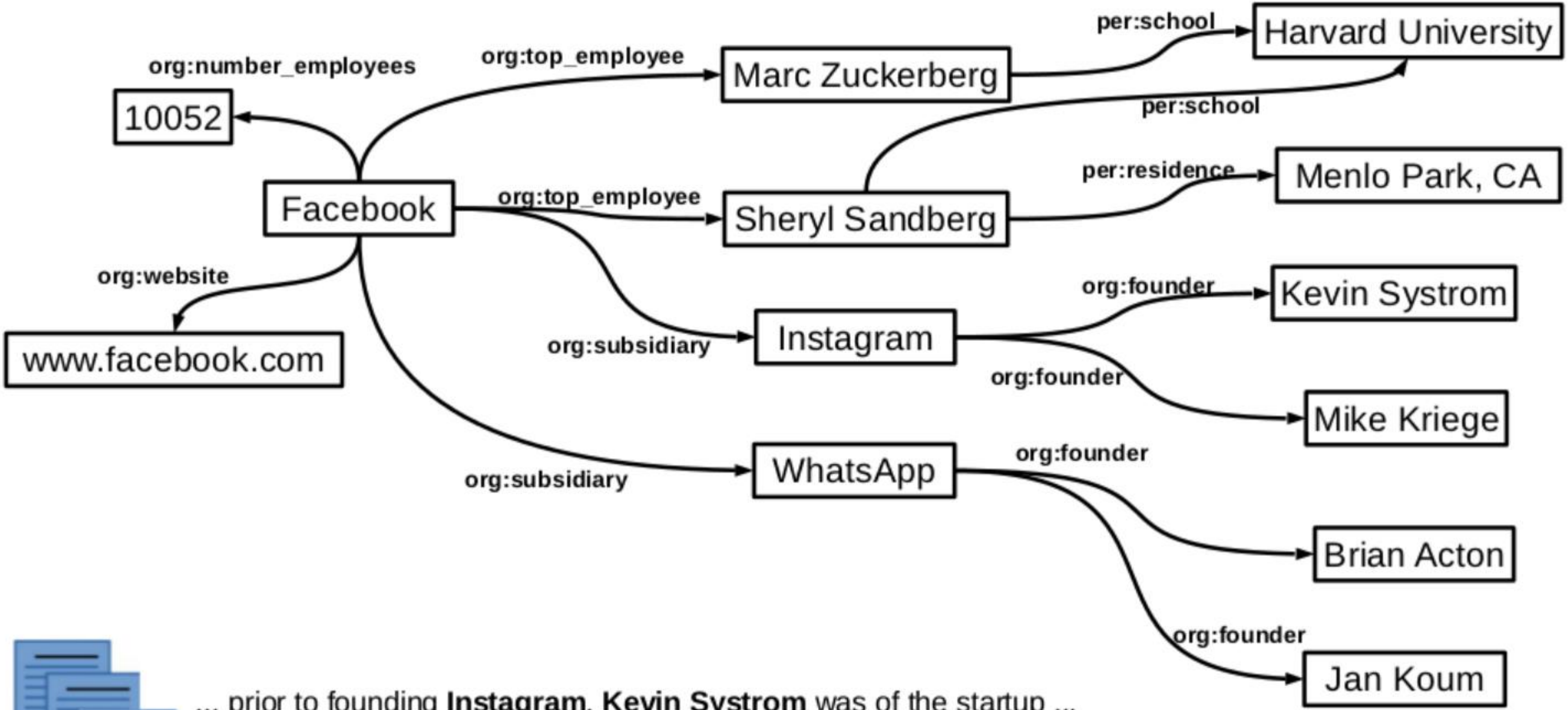
- Most information about real world is unstructured.
  - *"At the age of 19, Martin Luther entered the University of Erfurt."*
    "On 2 July 1505 he was returning to Erfurt after visiting his parents in Mansfeld."

  ⇒ Did Martin Luther live in Erfurt?

- Turning unstructured data into structured form:

  Automated knowledge base population (KBP)

  ⇒ `lived_in(M_Luther, Erfurt) 0.8942`



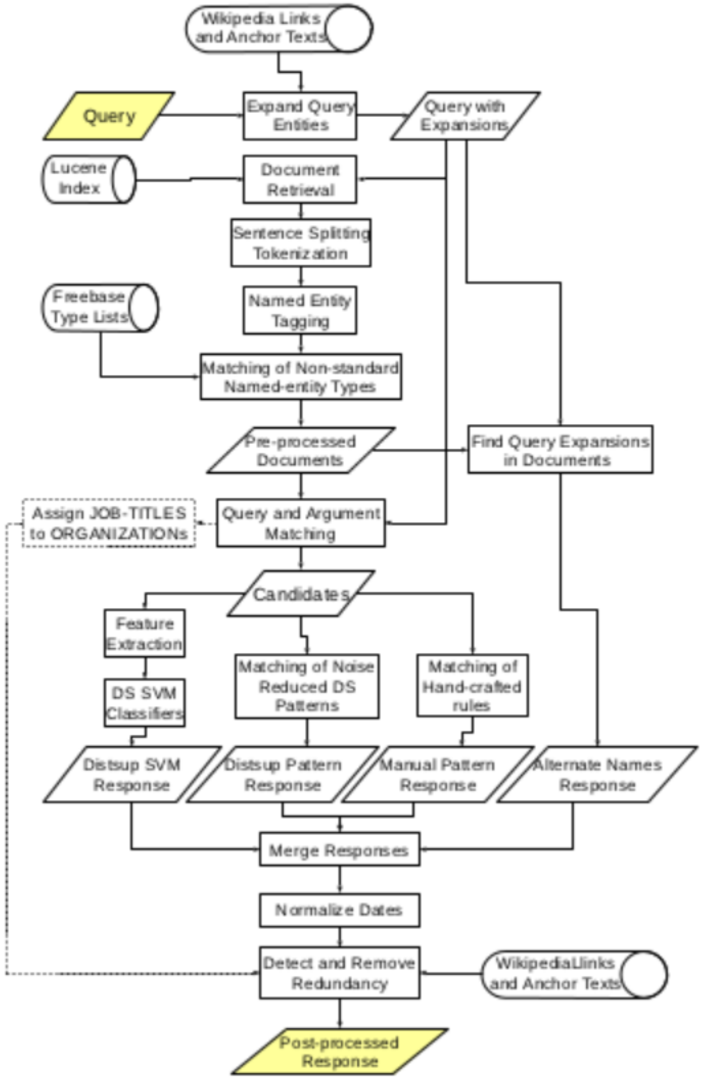
... prior to founding **Instagram**, **Kevin Systrom** was of the startup ...
... **Mike Krieger** co-founded **Instagram** with **Kevin Systrom** …
... reminiscent of **Instagram**'s parent company **Facebook Inc.** ...
... the $19 billion buyout of **Whatsapp** by **Facebook** ...

Why more structured data?

- Aggregate and combine information:
  - **Computational social science:**
    Detecting real world political events and trends in society
    [O'Connor, 2013, 2017]
  - **Science, e.g. Bio-informatics:**
    Extracting genome and protein interactions from research publications [Krallinger et al., 2017]
  - **Market research:**
    Extracting typical use-cases of food and products
    [Wiegand et al., 2014].
- Query structured data in dialogue systems:
  - E.g. Flight information [Seneff et al. 1991], **In-car assistants** [Madotto et al. 2018]

Traditional relation extraction

- Complex retrieval + filtering pipelines
- **Identify entities**, then predict relation
- Problems with traditional approach
  - Tagging errors, nested entities, type granularity
  - ***Non-standard entity types*** (≠PER,LOC,ORG,…)



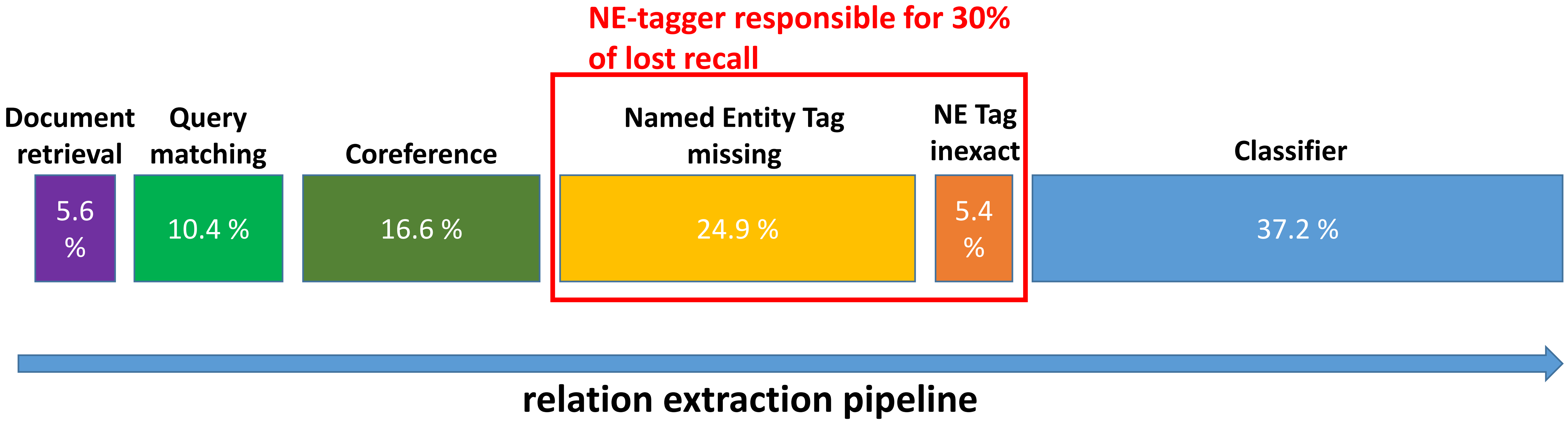
Problems with traditional approach

- *“[Popular Kabul]$_{ORG}$ lawmaker [Ramazan Bashardost]$_{PER}$, who camps out in a tent near parliament ...”*
  city-of-residence ?

- *“[Haig]$_{PER}$ attended the [US Army]$_{ORG}$ academy at [West Point]$_{LOC}$ ...”*
  school-attended ?

- *“[Michael Sandy]$_{PER}$ died after being [struck by a car]$_{DEATH\_CAUSE}$ as he ran from ...”*
  cause-of-death ?

Recall lost in pipeline



Relation extraction for non-standard types

- "Neural Architectures for Open-Type Relation Argument Extraction" [Roth, Conforti, Poerner, Karn, Schütze. NLE 2018]
- **Problem:** Named entity recognition
- **Solution:**
  - Relation prediction without NE tagger
  - Any subspan can be relation argument
  - **No restriction on argument types**

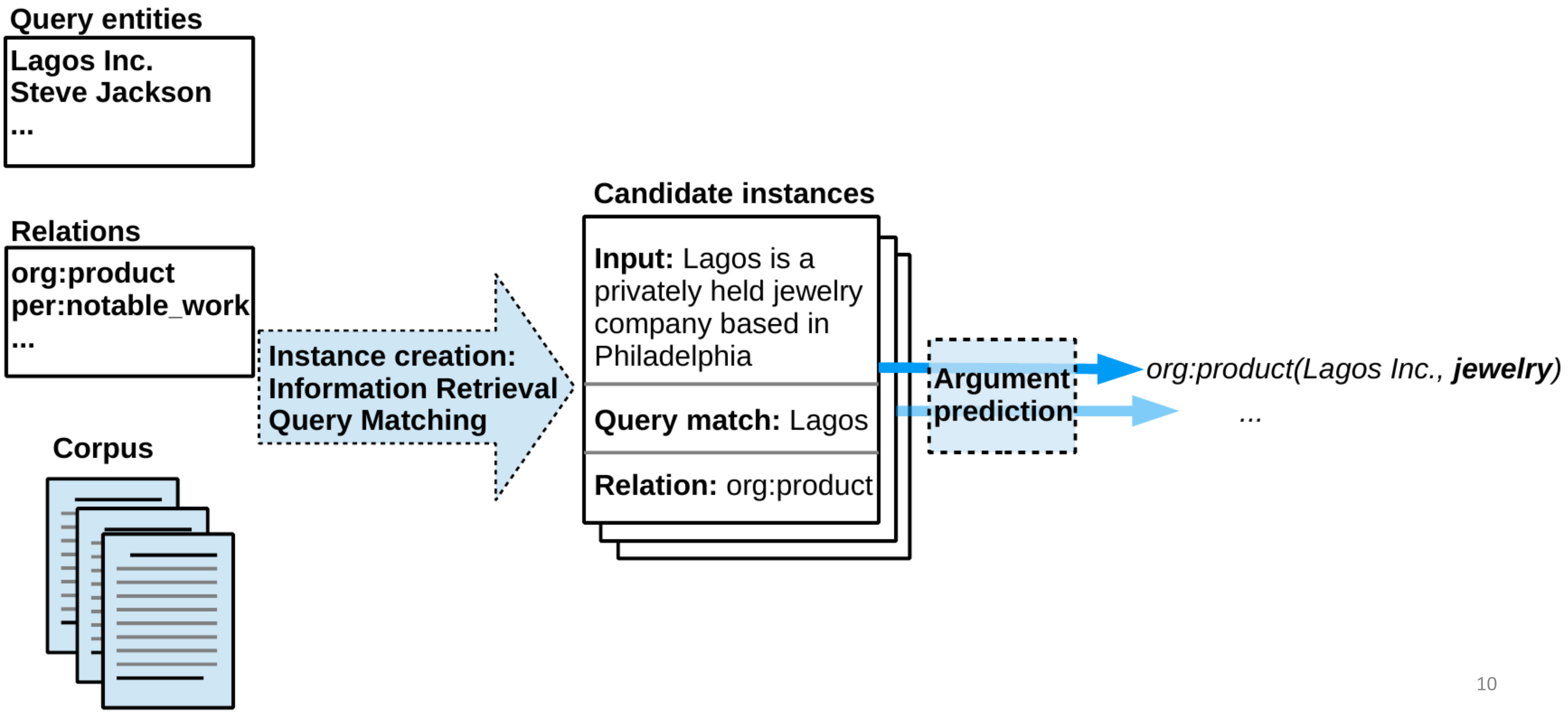
Query-driven relation prediction

Query: „Alexander Haig“

Context: „Haig attended the US army academy at Westpoint.“

- Traditional approach:
  - *“[Haig]$_{Query}$ attended the [US Army]$_{Answer}$ academy at West Point ...”*
    school-attended ➔ Yes / No?
  - *“[Haig]$_{Query}$ attended the US Army academy at [West Point]$_{Answer}$ ...”*
    born-in ➔ Yes / No?
  - **…**
- Proposed approach:
  - *“[Haig]$_{Query}$ attended the US Army academy at West Point ...”*
    school-attended ➔ Answer?
    born-in ➔ Answer?
    …

Query-driven argument extraction



Model

Query-driven argument extraction

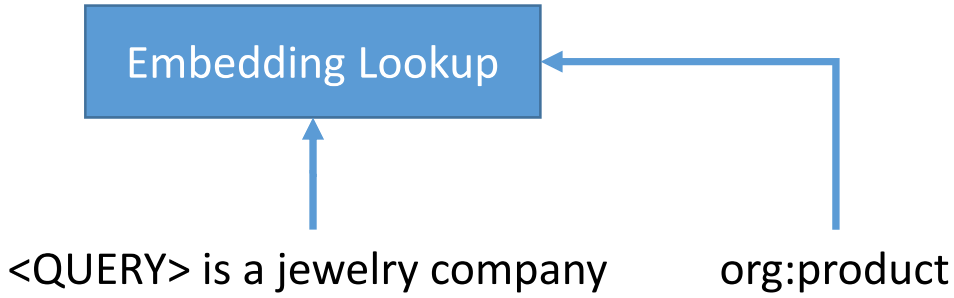
Lagos is a jewelry company org:product

Query-driven argument extraction

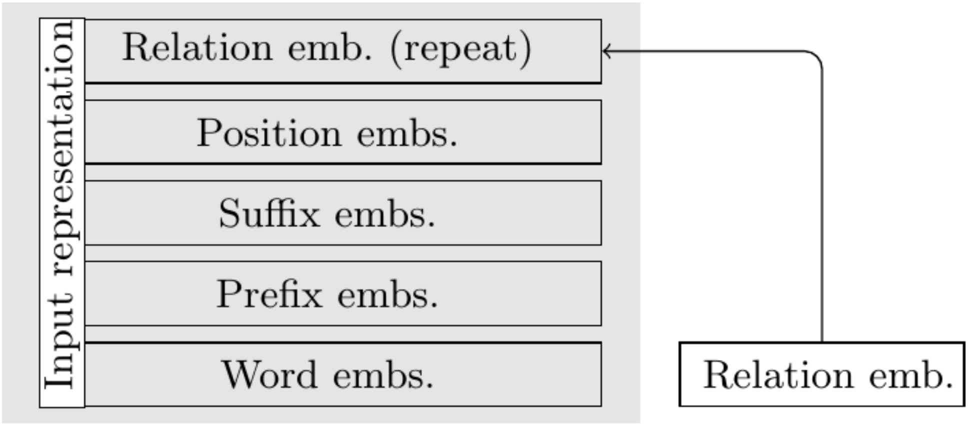
LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

<QUERY> is a jewelry company org:product

Query-driven argument extraction

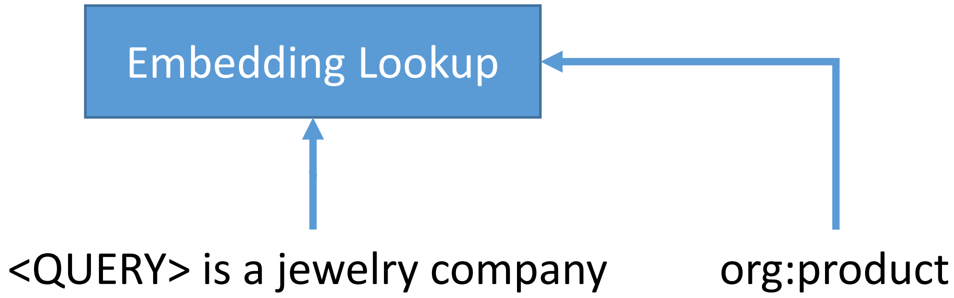


Query-driven argument extraction

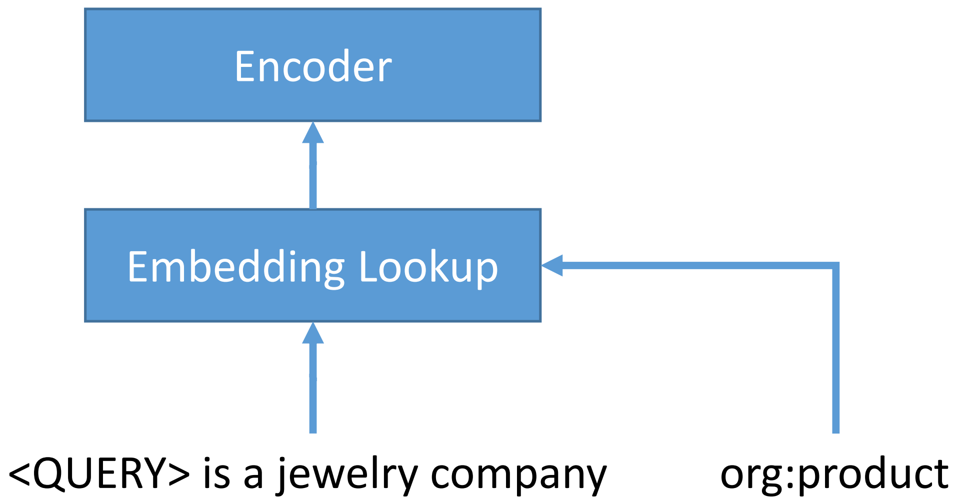


<QUERY> is a jewelry company org:product

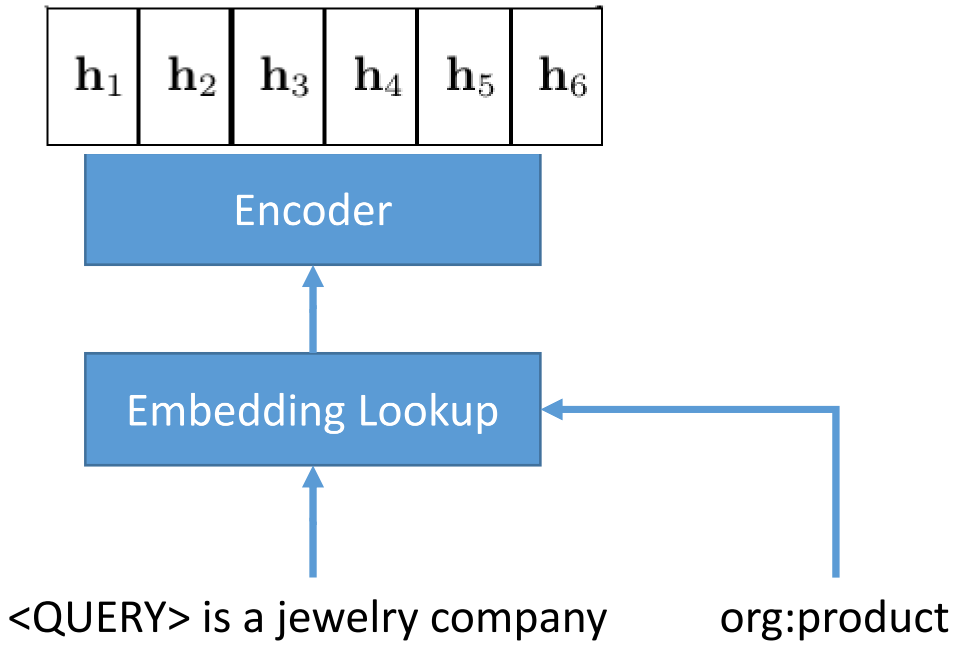
Query-driven argument extraction



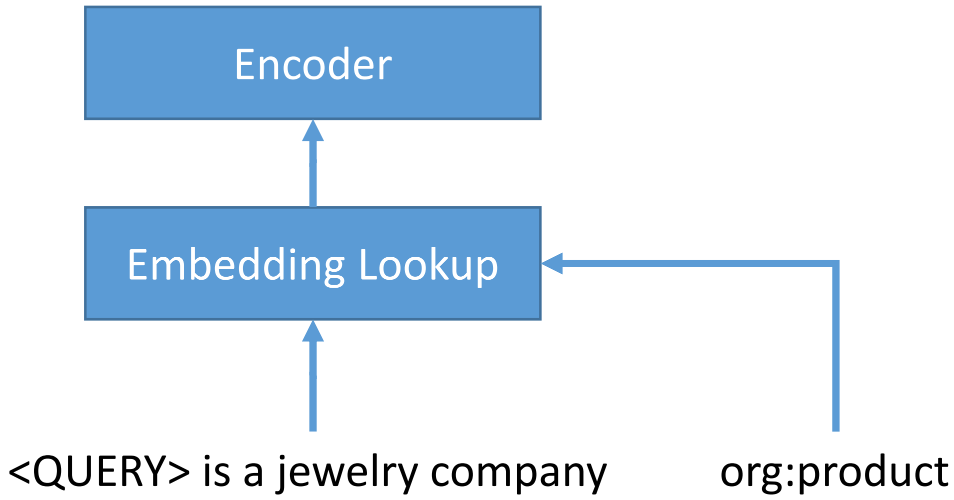
Query-driven argument extraction



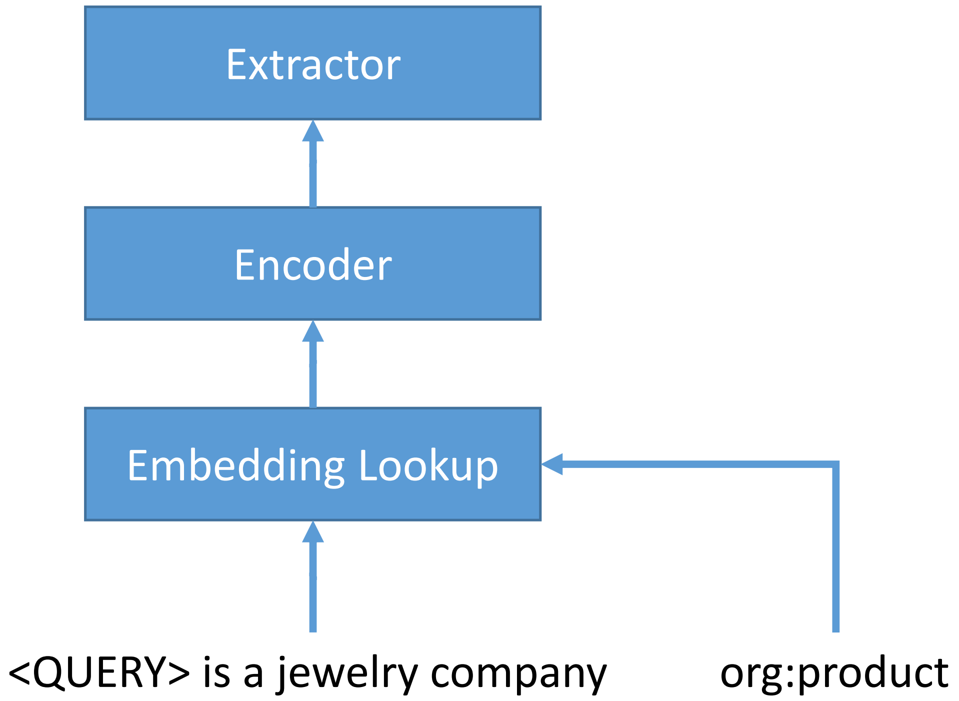
Query-driven argument extraction



Query-driven argument extraction

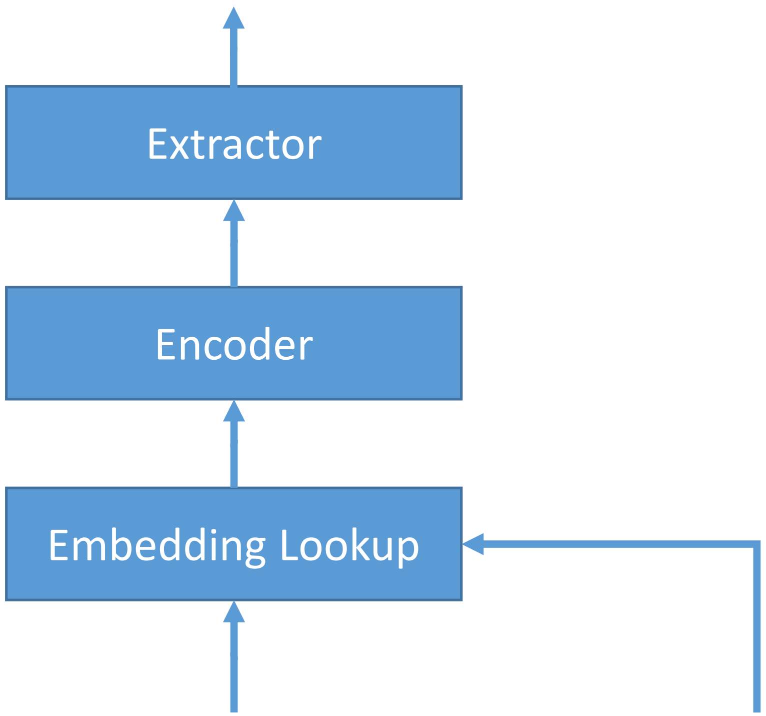


Query-driven argument extraction



Query-driven argument extraction

Predicted argument: “jewelry”



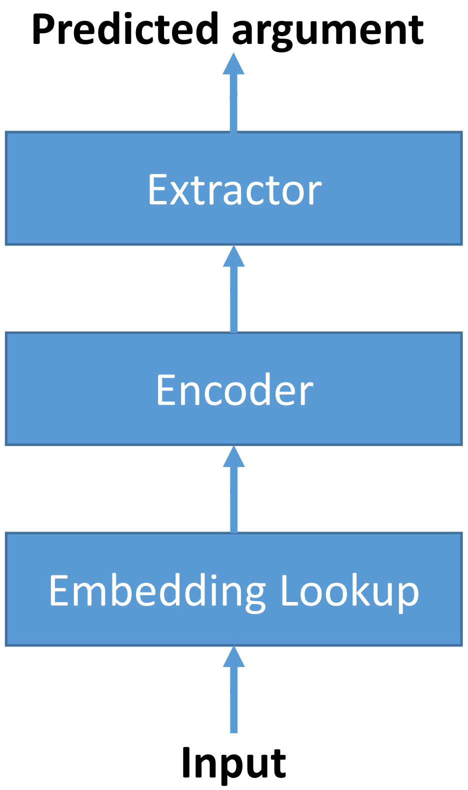
<QUERY> is a jewelry company   org:product

Encoder stage

- Encode candidate sentence into sequence of vectors.

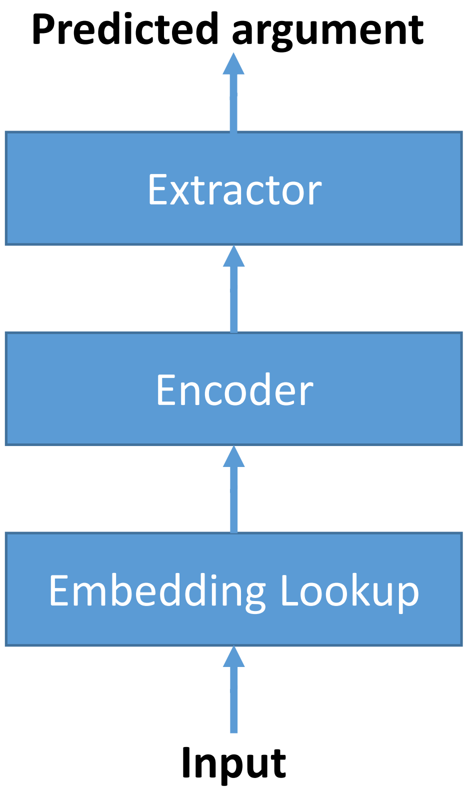
$\mathbf{h}_1$	$\mathbf{h}_2$	$\mathbf{h}_3$	$\mathbf{h}_4$	$\mathbf{h}_5$	$\mathbf{h}_6$	...

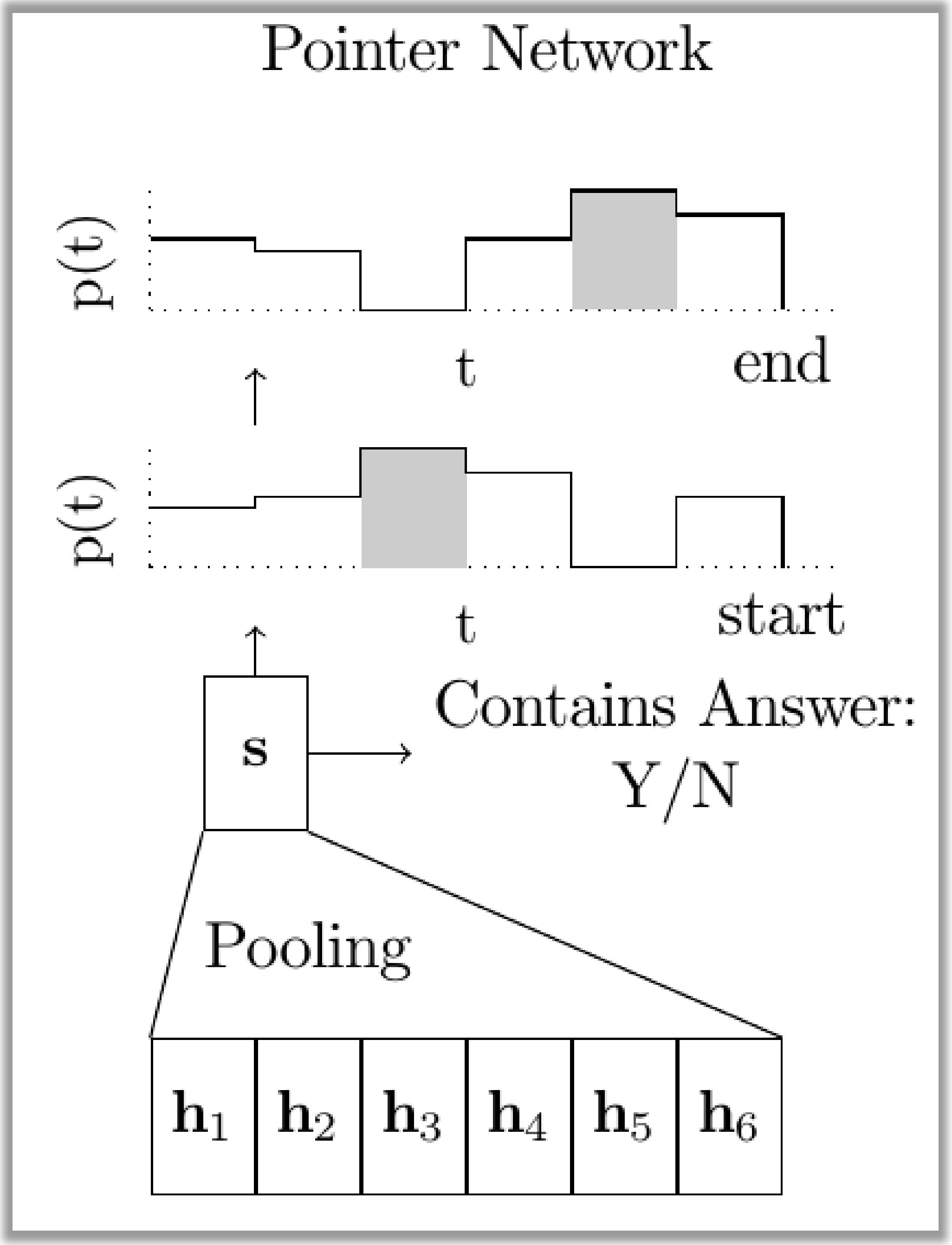
- Variants:
  - Bi-Directional Gated Recurrent Units (RNN) [Chung, 2014]
    - Standard for encoding sequences
    - **Inductive bias:** global with local bias
  - Convolutional neural networks (CNN) [Collobert, 2011]
    - Efficient processing
    - **Inductive bias:** local
  - Self-attention/Google Transformer (ATTN) [Vaswani, 2017]
    - Relatively recently proposed sequence encoder
    - Interaction with non-transformer layers?
    - **Inductive bias:** weak/global



Extractor stage

- Select subspan (relational argument)
- Variants:
  - Pointer network [Vinyals, 2015]
  - Table filling [Miwa, 2014]
  - Conditional random fields tagger (CRF) [Lample, 2016]

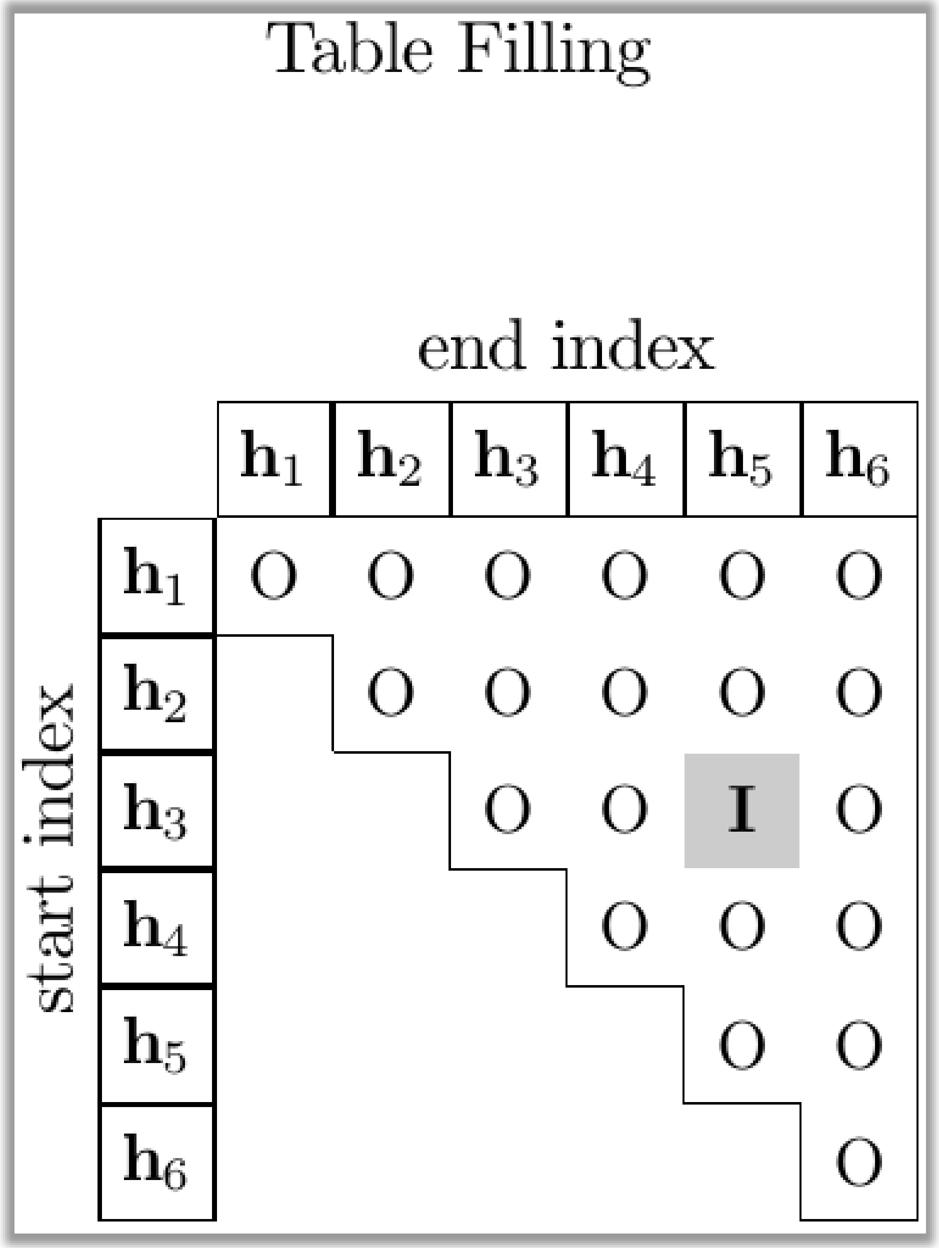




- Predict start position, then end position
- Predictions dependent, not joint!
- Many deep QA models are pointer networks

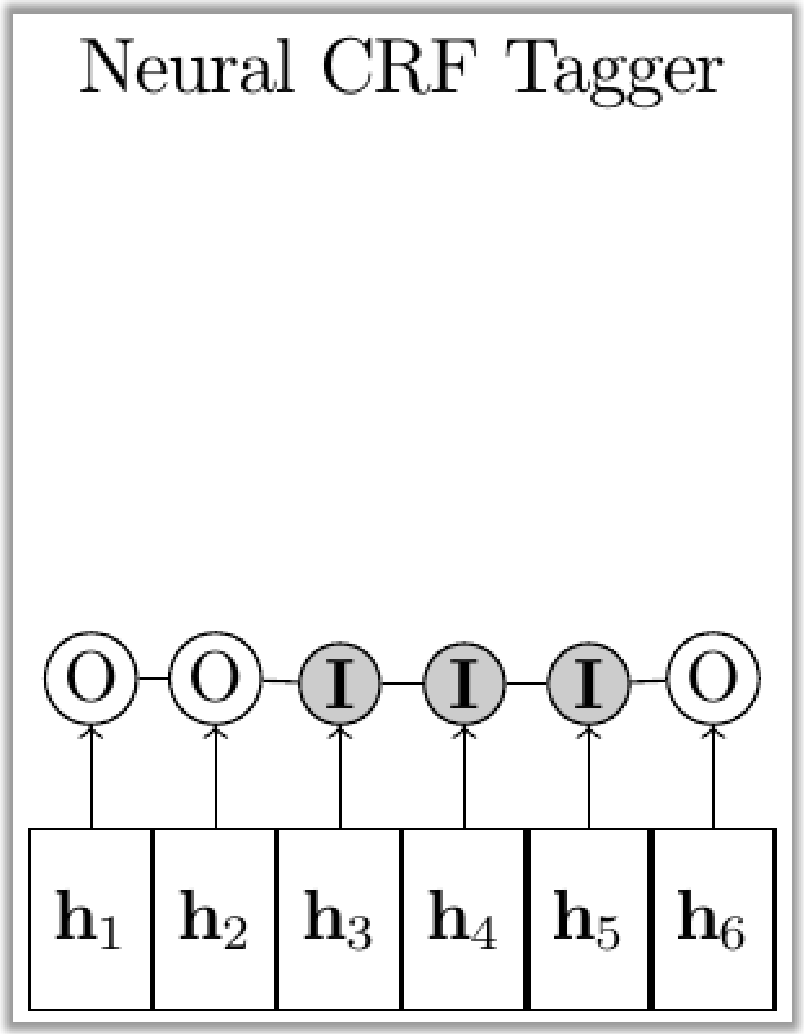
$$\bar{\mathbf{s}} = ReLU(W^s Pool(H))$$

$$p(start = i) = \text{softmax}(MLP([\bar{\mathbf{s}}; \mathbf{h}_i]))$$



- Decide for all pairs of start/end positions
- ~ joint version of pointer network
- Large number of negative cells

$$p(is\_answer = True | start = i, end = j)$$
$$= \sigma([\mathbf{h}_i; \mathbf{h}_j]^T \mathbf{w}^{(table)})$$



- Mark subsequence with I-tags
- Optimize global score
  - Local label scores (s)
  - Label compatibility (A)

$$s(H, \mathbf{y}) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} s_{i, y_i}$$

Data set

Data set

- First relation extraction data set with focus on non-standard types
    - entities ➔ concepts
- Requirements for selecting relations:
    - Missing argument has **non-standard type**. ~~location, person, organization,…~~
    - **Open class**. Wide range of admissible values (>1000). ~~gender,…~~
    - **Substantial coverage.** > 10000 facts in Wikidata for relation.
- Distant supervision from WikiData and Wikipedia
    - SPARQL
    - Elasticsearch
    - Entity expansion

relation	id	#sentences
per:occupation	P106	57693
per:position_held	P39	47386
per:conflict	P607	20575
per:notable_work	P800	18826
per:participant_of	P1344	14646
per:award_received	P166	13330
per:field_of_work	P101	13059
org:industry	P452	12352
per:noble_family	P53	9260
per:ethnic_group	P172	7169
org:product	P1056	6482
gpe:office	P1313	3781

Relations (examples)

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

`per:conflict`	It is named for [*Henry Knox*]$_Q$, an [*American Revolutionary War*]$_A$ general .
`per:notable_work`	In the [*Steve Jackson*]$_Q$ Games card game [*Munchkin*]$_A$, there is a card called " Dwarf Tossing " .
`per:field_of_work`	While teaching at Berkeley , [*John Harsanyi*]$_Q$ did extensive research in [*game theory*]$_A$.
`per:noble_family`	Stefan was the son of Lazar and his wife [*Milica*]$_Q$, a lateral line of [*Nemanjić*]$_A$.
`org:product`	[*Lagos*]$_Q$ is a privately held American [*jewelry*]$_A$ company

Experiments

Comparison: all combinations

	Pointer Network	Table Filling	Neural CRF
RNN	78.99	**79.64**	**81.86**
CNN	79.41	79.11	**79.61**
Self-Attention	74.49	75.89	74.35

- F1-scores
- Best encoder / extractor: RNN / CRF
- Self-attention disappoints

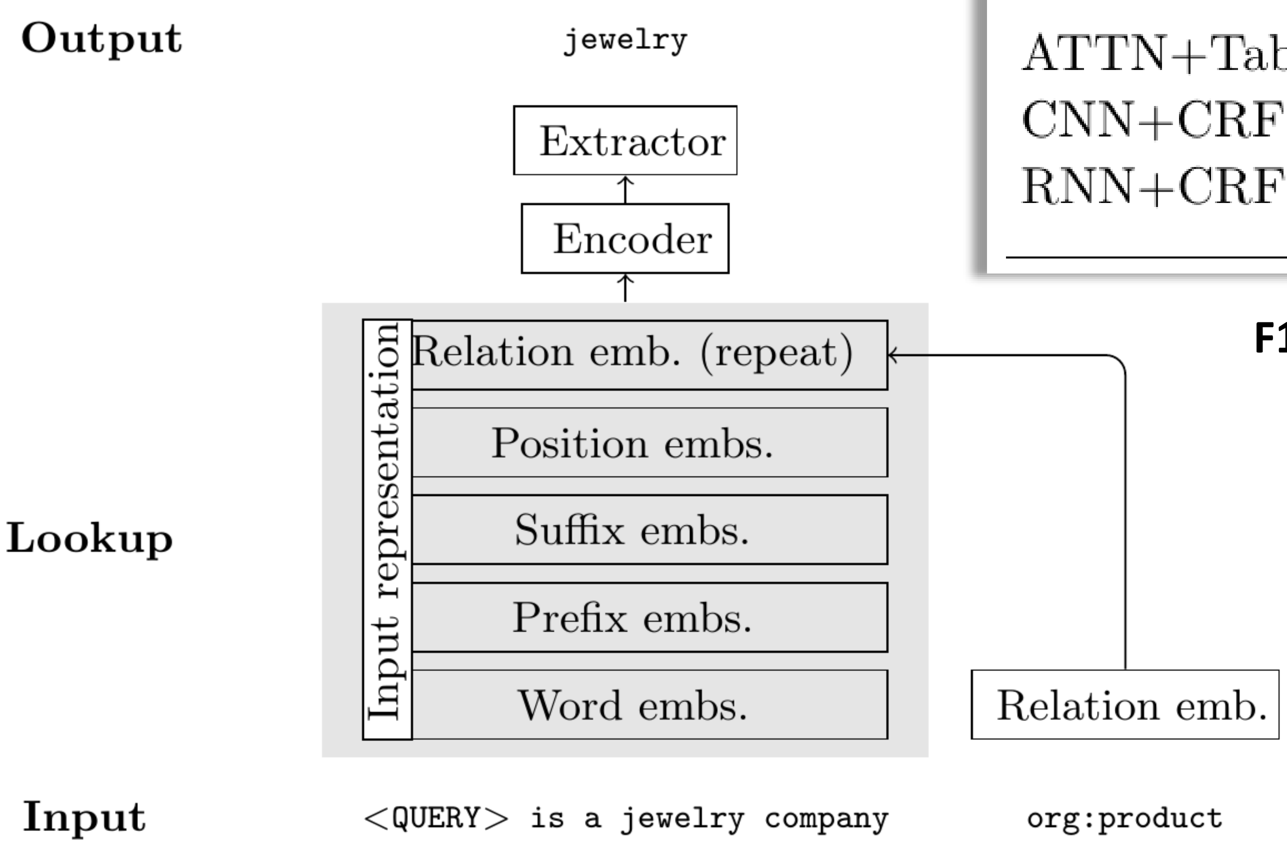
Baselines

- Bi-Directional Attention-Flow (**BiDAF** / AllenAi, Seo et al. 2017)
  - Neural question answering model
  - Pointer mechanism
  - For our task:
    Relation is 1-word question („org:product ?“)
- Position-aware Attention (**PosAtt** / Stanford, Zhang et al. 2017)
  - Neural relation classification model
  - Predicts relation given marked candidate arguments
  - For our task :
    Use answers from training data to match answer candidates in dev/test.

Comparison with baselines

	Prec	Rec	F1
BiDAF	70.86	78.76	74.60
PosAtt	**83.65**	$\boxed{72.11}$	77.45
CNN / CRF	82.59	76.84	79.61
RNN / Table	77.92	**81.44**	79.64
RNN / CRF	82.53	$\boxed{81.19}$	**81.86**

Ablation Analysis: Input encoding



Architecture	Word	Affix	Position	Query	Relation
ATTN+Table	0.16	1.73	3.54	4.89	50.76
CNN+CRF	2.63	0.16	0.16	2.87	74.06
RNN+CRF	2.73	0.05	-0.29	3.71	79.99

F1 improvement when (re-)adding embedding

Examples

- " ***The Emperor 's New Clothes*** " is a Danish fairy tale written by **[Hans Christian Andersen]$_{\textbf{Query}}$** and first published in 1837 .
  - relation: **per:notable_work**
  - gold answer: **The Emperor 's New Clothes**
  - predicted: **The Emperor 's New Clothes**
- Lucas won the 1977 ***Academy Award for Film Editing*** with **[Richard Chew]$_{\textbf{Query}}$** and Paul Hirsch for her work editing `` Star Wars . "
  - relation: **per:award_received**
  - gold answer: **Academy Award for Film Editing**
  - predicted: **Academy Award for Film Editing**

Example: wrong span

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

- North Star (anti-slavery newspaper) North Star was a nineteenth-century ***anti-slavery*** newspaper published from the Talman Building in Rochester , New York by ***abolitionist*** **[Frederick Douglass]$_{\textbf{Query}}$**.
  - relation: **per:field_of_work**
  - gold answer: **anti-slavery**
  - predicted: **abolitionist**

Example: missed answer

- Game Show Network Game Show Network (GSN) is an American digital cable and ***satellite television*** channel that is owned as a joint venture between Sony Pictures Television (owning a controlling 58 % interest) and **[AT & T]**$_{\text{Query}}$ Entertainment Group (holding a 42 % ownership stake) .
  - relation: **org:product_material_produced**
  - gold answer: **satellite television**
  - predicted: **-**

„End-to-end“

- We successfully removed the NE-tagger.
- OK, but what about the rest of the pipeline?
- How far is it reasonable to go? How many IR-steps to replace by deep models?
  - Embed the web for each query?
  - If deep, then
    - how wide? (how many instances)
    - how deep? (interactions modeled)
  - „Deep“ re-rankers ⇔ reasoning with memory networks
- Needs to be carefully explored for each task!
- Interesting: very deep reasoning on limited amount of retrieved instances

Conclusion

- First work to focus on non-standard entities in relation extraction
  - Data set with 12 relations for non-standard types
- Replaced named entity tagger by deep argument extraction model
- Competitive neural encoder-extractor architecture
  - **RNN**, CNN, Self-Attention
  - Pointer, Table filling, **CRF**
- What's the best place for deep learning in pipelined architectures?

Thank you!
Questions?