Zero-Shot Relation Extraction via Reading Comprehension

OMER LEVY, MINJOON SEO, EUNSOL CHOI, AND LUKE ZETTLEMOYER

PUBLISHED IN THE PROCEEDINGS OF THE 21ST CONFERENCE ON COMPUTATIONAL NATURAL LANGUAGE LEARNING (CONLL 2017), AUGUST 2017.

PRESENTER: KEVIN XIE

Zero-Shot **Relation Extraction** via Reading Comprehension

WHAT IS RELATION EXTRACTION?

A relation connects an entity to a concept in unstructured text

Sentence **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

A relation connects an entity to a concept in unstructured text

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Relation **R**: Election_Year

A relation connects an entity to a concept in unstructured text

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Relation **R**: Election_Year

Entity **e**: Barack Obama

A relation <u>connects</u> an <u>entity</u> to a <u>concept</u> in <u>unstructured text</u>

Sentence **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

Relation **R**: Election_Year

Entity **e**: Barack Obama

Concept **a**: 2008

A relation connects an entity to a concept in unstructured text

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Relation **R**: Election_Year

Entity **e**: Barack Obama

Concept **a**: 2008

Functionally: Given **s**, **R**(**e**, **a**).

- Election_Year(Barack Obama, 2008)

Relation Extraction (RE) tries to construct **R(e, a)** given **s** and some combination of **R, e, a**

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Relation Extraction (RE) tries to construct **R(e, a)** given **s** and some combination of **R, e, a**

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Traditionally: Given **e, a**, and $\bar{\mathbf{R}} = \{\mathbf{R_1}, \mathbf{R_2}, \ldots \mathbf{R_N}\}$. Map(**e**, **a**) ➔ $\mathbf{R} \in \bar{\mathbf{R}}$

- **e**: Barrack Obama, **a:** 2008 ➔ **R**: Election_Year

Relation Extraction (RE) tries to construct **R(e, a)** given **s** and some combination of **R, e, a**

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Traditionally: Given **e, a**, and $\bar{\mathbf{R}} = \{\mathbf{R_1}, \mathbf{R_2}, \ldots \mathbf{R_N}\}$. Map(**e**, **a**) ➔ $\mathbf{R} \in \bar{\mathbf{R}}$

- **e**: Barrack Obama, **a:** 2008 ➔ **R**: Election_Year

Sounds like Multi-class classification!

Relation Extraction (RE) tries to construct **R(e, a)** given **s** and some combination of **R, e, a**

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Traditionally: Given **e, a**, and $\bar{\mathbf{R}} = \{\mathbf{R_1}, \mathbf{R_2}, \dots \mathbf{R_N}\}$. Map(**e**, **a**) ➔ $\mathbf{R} \in \bar{\mathbf{R}}$

- **e**: Barrack Obama, **a:** 2008 ➔ **R**: Election_Year

Sounds like Multi-class classification!

Classic ML with Feature Extraction:

- Zelenko et al. 2002: Kernel methods for RE
- Jiang and Zhai 2007: Feature spaces for RE

DL RE

- Lin et al. 2016: CNN-based embeddings with Sentence-level Attention

Jiang, Jing, and ChengXiang Zhai. "A Systematic Exploration of the Feature Space for Relation Extraction." *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Association for Computational Linguistics, 2007, pp. 113–20. *ACLWeb*, https://www.aclweb.org/anthology/N07-1015.
Lin, Yankai, et al. "Neural Relation Extraction with Selective Attention over Instances." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, 2016, pp. 2124–33. *DOI.org (Crossref)*, doi:10.18653/v1/P16-1200.
Zelenko, Dmitry, et al. "Kernel Methods for Relation Extraction." *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - EMNLP '02*, vol. 10, Association for Computational Linguistics, 2002, pp. 71–78. *DOI.org (Crossref)*, doi:10.3115/1118693.1118703.

Relation Extraction (RE) tries to construct **R(e, a)** given **s** and some combination of **R, e, a**

Sentence **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Traditionally: Given **e, a**, and $\overline{\mathbf{R}} = \{\mathbf{R_1}, \mathbf{R_2}, \ldots \mathbf{R_N}\}$. Map(**e**, **a**) ➔ $\mathbf{R} \in \overline{\mathbf{R}}$

- **e**: Barrack Obama, **a:** 2008 ➔ **R**: Election_Year
- **N** is typically small
  - Zelenko et al. 2002: **N** = 2 (Binary Classification)
  - Lin et al. 2016: **N** = 52 + 1 NA

Jiang, Jing, and ChengXiang Zhai. "A Systematic Exploration of the Feature Space for Relation Extraction." *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Association for Computational Linguistics, 2007, pp. 113–20. *ACLWeb*, https://www.aclweb.org/anthology/N07-1015.

Lin, Yankai, et al. "Neural Relation Extraction with Selective Attention over Instances." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, 2016, pp. 2124–33. *DOI.org (Crossref)*, doi:10.18653/v1/P16-1200.

Zelenko, Dmitry, et al. "Kernel Methods for Relation Extraction." *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - EMNLP '02*, vol. 10, Association for Computational Linguistics, 2002, pp. 71–78. *DOI.org (Crossref)*, doi:10.3115/1118693.1118703.

Relation Extraction (RE) tries to construct **R(e, a)** given **s** and some combination of **R, e, a**

Sentence **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

Traditionally: Given **e, a**, and $\bar{\mathbf{R}} = \{\mathbf{R_1}, \mathbf{R_2}, \dots \mathbf{R_N}\}$. Map(**e**, **a**) ➔ $\mathbf{R} \in \bar{\mathbf{R}}$

- **e**: Barrack Obama, **a:** 2008 ➔ **R**: Election_Year
- **N** is typically small
  - Zelenko et al. 2002: **N** = 2 (Binary Classification)
  - Lin et al. 2016: **N** = 52 + 1 NA

Levy et al: Given $\mathbf{R} \in \bar{\mathbf{R}}$, and **e**. Map(**R**, **e**) ➔ **a**

- **N** = 120

Jiang, Jing, and ChengXiang Zhai. “A Systematic Exploration of the Feature Space for Relation Extraction.” *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Association for Computational Linguistics, 2007, pp. 113–20. *ACLWeb*, https://www.aclweb.org/anthology/N07-1015.
Lin, Yankai, et al. “Neural Relation Extraction with Selective Attention over Instances.” *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, 2016, pp. 2124–33. *DOI.org (Crossref)*, doi:10.18653/v1/P16-1200.
Zelenko, Dmitry, et al. “Kernel Methods for Relation Extraction.” *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - EMNLP ’02*, vol. 10, Association for Computational Linguistics, 2002, pp. 71–78. *DOI.org (Crossref)*, doi:10.3115/1118693.1118703.

Levy et al. tries to identify **a** (the "answer") in **R(e, a)** given **s**

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Levy et al. tries to identify **a** (the "answer") in **R(e, a)** given **s**

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

Levy et al. tries to identify **a** (the "answer") in **R(e, a)** given **s**

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

Office(Barack Obama, **a**) ➔ **a** = president

Levy et al. tries to identify **a** (the "answer") in **R(e, a)** given **s**

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

Office(Barack Obama, **a**) ➔ **a** = president

Spouse(Barack Obama, **a**) ➔ ?

Levy et al. tries to identify **a** (the "answer") in **R(e, a)** given **s**

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

Office(Barack Obama, **a**) ➔ **a** = president

Spouse(Barack Obama, **a**) ➔ **a** = N/A

Zero-Shot **Relation Extraction** via Reading Comprehension

WHAT IS ZERO-SHOT RELATION EXTRACTION?

Zero-shot RE "defines new relations 'on the fly', *after* the model has already been trained"

Suppose model **M** is trained on **N** relation types, forming training set $\overline{R}_{\mathbf{N}}$.

Zero-shot RE "defines new relations 'on the fly', *after* the model has already been trained"

Suppose model **M** is trained on **N** relation types, forming training set $\overline{R}_{\mathbf{N}}$.

How does **M** perform on a new, unseen type $\mathbf{R_{N+1}}$?

Zero-shot RE presents new relation types at test time

Suppose **M** was trained on **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

The training set included the following **R(e, ?)**:

- Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden
- Office(Barack Obama, **a**) ➔ **a** = president
- Spouse(Barack Obama, **a**) ➔ **a** = N/A

Zero-shot RE presents new relation types at test time

Suppose **M** was trained on **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

The training set included the following **R(e, ?)**:

- Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden
- Office(Barack Obama, **a**) ➔ **a** = president
- Spouse(Barack Obama, **a**) ➔ **a** = N/A

The zero-shot test set could include:

- Action(Barack Obama, **a**) ➔ **a** = ran for president

Why do we want zero-shot RE?

There are potentially infinite sentences, each with their own different **R**(**e**, **a**)

Why do we want zero-shot RE?

There are potentially infinite sentences, each with their own different **R**(**e**, **a**)

It is impossible to annotate them all for supervised training/testing

Why do we want zero-shot RE?

There are potentially infinite sentences, each with their own different **R**(**e**, **a**)

It is impossible to annotate them all for supervised training/testing

Any model that is useable in the real-world must be able to generalize to unseen examples

Zero-Shot **Relation Extraction** via Reading Comprehension

HOW DOES READING COMPREHENSION FACTOR INTO ZERO-SHOT RELATION EXTRACTION?

Reading Comprehension is Question-Answering (QA) of (unstructured) text

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Reading Comprehension is Question-Answering (QA) of (unstructured) text

s: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

Q = {“Who was Obama’s running mate?”, “What did Obama run for in 2008?”, “Who is Obama’s Spouse?”}

A = {“Joe Biden”, “President”, “N/A”}

Reading Comprehension is Question-Answering (QA) of (unstructured) text

s: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

Q = {“Who was Obama’s running mate?”, “What did Obama run for in 2008?”, “Who is Obama’s Spouse?”}

A = {“Joe Biden”, “President”, “N/A”}

Hmm, these answers seem familiar…

Levy et al. tries to identify **a** (the "answer") in **R(e, a)** given **s**

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

Office(Barack Obama, **a**) ➔ **a** = president

Spouse(Barack Obama, **a**) ➔ **a** = N/A

RE can be a QA problem!

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

RE:

- Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden
- Office(Barack Obama, **a**) ➔ **a** = president
- Spouse(Barack Obama, **a**) ➔ **a** = N/A

QA:

- **Q =** {"Who was Obama's running mate?", "What did Obama run for in 2008?",
  "Who is Obama's Spouse?"}
- **A =** {"Joe Biden", "President", "N/A"}

Formulating RE under the Reading Comprehension paradigm

Levy et al. created question sets for relations

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Levy et al. created question sets for relations

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

- Who did Barack Obama run with in 2008?
- Who was Barack Obama's running mate in 2008?

Levy et al. created question sets for relations

s: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

Running_Mate(Barack Obama, **a**) ➔ **a** = Joe Biden

- Who did Barack Obama run with in 2008?
- Who was Barack Obama’s running mate in 2008?

Office(Barack Obama, **a**) ➔ **a** = President

- What did Barack Obama run for in 2008?
- For which office did Barack Obama run for?

Making specific questions for relations is expensive. Templates can reduce the cost

Relation	Question Template
$educated\_at(x, y)$	Where did x graduate from? In which university did x study? What is x's alma mater?
$occupation(x, y)$	What did x do for a living? What is x's job? What is the profession of x?
$spouse(x, y)$	Who is x's spouse? Who did x marry? Who is x married to?

Figure 1: Common knowledge-base relations defined by natural-language question templates.

Creating question templates ("Querifying") was crowd-sourced

Annotators were presented with masked example sentences

- "x ran for president in 2008 with Joe Biden as his running mate"

Creating question templates ("Querifying") was <u>crowd-sourced</u>

Annotators were presented with masked example sentences

- "x ran for president in 2008 with Joe Biden as his running mate"

Possible answers **a** for some relation **R(x, a)** were underlined

- "x ran for president in 2008 with <u>Joe Biden</u> as his running mate"

Creating question templates ("Querifying") was crowd-sourced

Annotators were presented with masked example sentences

- "x ran for president in 2008 with Joe Biden as his running mate"

Possible answers **a** for some relation **R(x, a)** were underlined

- "x ran for president in 2008 with Joe Biden as his running mate"

Annotators create questions about entity **x** with answer **a**.

- "Who was x's running mate?"

Creating question templates ("Querifying") was crowd-sourced

Annotators were presented with masked example sentences

- "x ran for president in 2008 with Joe Biden as his running mate"

Possible answers **a** for some relation **R(x, a)** were underlined

- "x ran for president in 2008 with Joe Biden as his running mate"

Annotators create questions about entity **x** with answer **a**.

- "Who was x's running mate?"

In total, 1192 templates across 120 relations

Negative examples must be accounted for in training and testing

s: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

- Spouse(Barack Obama, **a**) ➔ **a** = N/A

Negative examples must be accounted for in training and testing

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

- Spouse(Barack Obama, **a**) ➔ **a** = N/A

Solution: mismatch a question **q** for one relation with a sentence **s** that expresses another relation

- Who is Barack Obama's spouse?

Performing RE with Reading Comprehension

Levy et al. used BiDAF (Seo et al. 2016) to identify answer spans

BiDAF returns confidence scores for each potential start and end of an answer

s: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”

q: “Who was Barack Obama’s running mate?”

	Barack	Obama	Ran	For	President	In	2008	With	Joe	Biden	As	His	Running	Mate
Start	6	4	1	2	1	3	2	1	**10**	6	2	1	3	1
End	2	7	1	3	2	2	1	2	4	**10**	2	2	1	3

*Values here are for illustration only. They were arbitrarily set and do not reflect any data

Seo, Minjoon, et al. “Bidirectional Attention Flow for Machine Comprehension.” *ArXiv:1611.01603 [Cs]*, June 2018. *arXiv.org*, http://arxiv.org/abs/1611.01603.

Confidence scores can be transformed into pseudo-probabilities

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

q: "Who was Barack Obama's running mate?"

	Barack	Obama	Ran	For	President	In	2008	With	Joe	Biden	As	His	Running	Mate
P(start)	0.22	0.1	0.01	0.02	0.01	0.04	0.02	0.01	**0.4**	0.22	0.02	0.01	0.045	0.01
P(end)	0.02	0.31	0.01	0.05	0.02	0.02	0.01	0.02	0.08	**0.4**	0.02	0.02	0.01	0.05

*Values here are for illustration only. They were arbitrarily set and do not reflect any data

Trained bias b is used to indicate confidence of no answer

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

q: "Who was Barack Obama's running mate?"

	Barack	Obama	Ran	For	President	In	2008	With	Joe	Biden	As	His	Running	Mate	b
P(start)	0.22	0.1	0.01	0.02	0.01	0.04	0.02	0.01	**0.4**	0.22	0.02	0.01	0.045	0.01	0.12
P(end)	0.02	0.31	0.01	0.05	0.02	0.02	0.01	0.02	0.08	**0.4**	0.02	0.02	0.01	0.05	0.16

*Values here are for illustration only. They were arbitrarily set and do not reflect any data

Pick the span of text with largest probability as the answer

s: "Barack Obama ran for president in 2008 with Joe Biden as his running mate."

q: "Who was Barack Obama's running mate?"

$$P(\boldsymbol{a} = span(i,j)) = softmax(P(start) * P(end))$$
$$P(\boldsymbol{a} = \emptyset) = P(b_{start}) * P(b_{end})$$

	Barack	Obama	Ran	For	President	In	2008	With	Joe	Biden	As	His	Running	Mate	b
P(start)	0.22	0.1	0.01	0.02	0.01	0.04	0.02	0.01	**0.4**	0.22	0.02	0.01	0.045	0.01	0.12
P(end)	0.02	0.31	0.01	0.05	0.02	0.02	0.01	0.02	0.08	**0.4**	0.02	0.02	0.01	0.05	0.16

*Values here are for illustration only. They were arbitrarily set and do not reflect any data

Experimental Results

Levy et al. variations of RE by QA models

Single Template: only one $\mathbf{q} \in \mathbf{Q_R}$ is used for that $\mathbf{R}$

- $\mathbf{q}$ = "Who is X's running mate?" for all cases where $\mathbf{R}$ = running_mate appears

Levy et al. variations of RE by QA models

Single Template: only one $\mathbf{q} \in \mathbf{Q_R}$ is used for that **R**

- **q** = "Who is X's running mate?" for all cases where **R** = running_mate appears

Multiple Templates: any $\mathbf{q} \in \mathbf{Q_R}$ is used for that **R**

- **q** = "Who is X's running mate?" or **q** = "Who ran with X?" for cases where **R** = running_mate appears.

Levy et al. variations of RE by QA models

Single Template: only one $\mathbf{q} \in \mathbf{Q_R}$ is used for that $\mathbf{R}$

- **q** = "Who is X's running mate?" for all cases where **R** = running_mate appears

Multiple Templates: any $\mathbf{q} \in \mathbf{Q_R}$ is used for that $\mathbf{R}$

- **q** = "Who is X's running mate?" or **q** = "Who ran with X?" for cases where **R** = running_mate appears.

Question Ensemble: multiple $\mathbf{q} \in \mathbf{Q_R}$ is used for that $\mathbf{R}$

- Both **q** = "Who is X's running mate?" and **q** = "Who ran with X?" for cases where **R** = running_mate appears.

Comparing RE by QA with other models

Knowledge-base relation: Relation indicators ($\mathbf{R}_{13}$) instead of questions

Natural Language relation: Relation names (running_mate) instead of questions

Random baseline: chooses random entity in sentence that is not in questions

Hewlett et al. 2016: RNN Labeler

Miwa and Bansal 2016: Relation Extractor

How does RE by QA perform on unseen entities?

Partition dataset along entities in questions

- Barack Obama in training only
  - **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate”
  - **q**: Who was Barack Obama’s Running mate?
- FDR in testing only
  - **s’**: “Roosevelt was elected president in 1933 during the Great Depression”
  - **q’**: “What year was Roosevelt elected?”

How does RE by QA perform on unseen entities?

Partition dataset along entities in question

- Barack Obama in training only
  - **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate"
  - **q**: Who was Barack Obama's Running mate?
- FDR in testing only
  - **s'**: "Roosevelt was elected president in 1933 during the Great Depression"
  - **q'**: "What year was Roosevelt elected?"

Sample 1M/1K/10K examples for Train/Dev/Test split

RE by QA performs well on unseen entities relative to competitors

RE by QA generalizes well when new entities are introduced for old relations

	Precision	Recall	F1
Random NE	11.17%	22.14%	14.85%
RNN Labeler	62.55%	62.25%	62.40%
Miwa & Bansal	96.07%	58.70%	72.87%
KB Relation	89.08%	91.54%	90.29%
NL Relation	88.23%	91.02%	89.60%
Single Template	77.92%	73.88%	75.84%
Multiple Templates	87.66%	91.32%	89.44%
Question Ensemble	88.08%	91.60%	89.80%

Table 1: Performance on unseen entities.

How does RE by QA perform on new templates (new questions)?

10 folds of train/dev/test with one question template for each relation held out for test set, and another for dev set

- **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate”

How does ER by QA perform on new templates (new questions)?

10 folds of train/dev/test with one question template for each relation held out for test set, and another for dev set

- **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate"
- $\mathbf{q_{train}}$: "Who was Barack Obama's running mate?"
- $\mathbf{q_{test}}$: "Who ran with Barack Obama?"
- $\mathbf{q_{dev}}$: "What was Barack Obama's running mate's name?"

How does ER by QA perform on new templates (new questions)?

10 folds of train/dev/test with one question template for each relation held out for test set, and another for dev set

- **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate"
- $\mathbf{q_{train}}$: "Who was Barack Obama's running mate?"
- $\mathbf{q_{test}}$: "Who ran with Barack Obama?"
- $\mathbf{q_{dev}}$: "What was Barack Obama's running mate's name?"

Sample **N** = 1K/10/50 examples per question template for train/dev/test

RE by QA generalizes to new templates

RE by QA experiences small performance decrease when new questions are asked for old relations

	Precision	Recall	F1
Seen	86.73%	86.54%	86.63%
Unseen	84.37%	81.88%	83.10%

Table 2: Performance on seen/unseen questions.

How does RE by QA perform on new, unseen relations (pure zero-shot)?

10 folds of train/dev/test partitioned along relations

- **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate"
- $\mathbf{R_{train}}$: running_mate(Barack Obama, **a**)
- $\mathbf{R_{test}}$: election_year(Barack Obama, **a**)
- $\mathbf{R_{dev}}$: office(Barack Obama, **a**)

How does RE by QA perform on new, unseen relations (pure zero-shot)?

10 folds of train/dev/test partitioned along relations

- **s**: "Barack Obama ran for president in 2008 with Joe Biden as his running mate"
- $\mathbf{R}_{\mathbf{train}}$: running_mate(Barack Obama, **a**)
- $\mathbf{R}_{\mathbf{test}}$: election_year(Barack Obama, **a**)
- $\mathbf{R}_{\mathbf{dev}}$: office(Barack Obama, **a**)

Partition 84/12/24 relations for train/dev/test

RE by QA beats competitors at pure zero-shot testing

RE by QA experiences significant decrease in performance on new relations, but is better than its competitors

	Precision	Recall	F1
Random NE	9.25%	18.06%	12.23%
RNN Labeler	13.28%	5.69%	7.97%
Miwa & Bansal	100.00%	0.00%	0.00%
KB Relation	19.32%	2.54%	4.32%
NL Relation	40.50%	28.56%	33.40%
Single Template	37.18%	31.24%	33.90%
Multiple Templates	43.61%	36.45%	39.61%
Question Ensemble	45.85%	37.44%	41.11%

Table 3: Performance on unseen relations.

RE by QA is affected by distractors in negative examples

Suppose the following **R(e, a)** problem:

- **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”
- **q**: “Who is Barack Obama married to?”
- Correct **a**: N/A
- Distractor **a’**: Joe Biden

RE by QA is affected by distractors in negative examples

Suppose the following **R(e, a)** problem:

- **s**: “Barack Obama ran for president in 2008 with Joe Biden as his running mate.”
- **q**: “Who is Barack Obama married to?”
- Correct **a**: N/A
- Distractor **a’**: Joe Biden

Analysis of random (negative) examples found:

- 35% contain distractors
- 1/7 error rate on negative examples with distractors
- 1/26 error rate on easier negative examples

Thoughts and Conclusions

RE by QA seems neat, but…

Question (Template) generation requires manual effort, and there are more questions than there are relations

- Expensive to create large (template) datasets ➔ poor scalability
- Cannot be easily translated to other languages ➔ no improvement in low resource languages

RE by QA seems neat, but…

Question (Template) generation requires manual effort, and there are more questions than there are relations

- Expensive to create large (template) datasets ➔ poor scalability
- Cannot be easily translated to other languages ➔ no improvement in low resource languages

RE by QA model performed well only <u>relative to compared models</u>

- ~40% F1 is far from human performance

Also,

What is the agreement between annotators, or a measure of crowdsourced validity?

- Authors only report that most question templates were unique

Also,

What is the agreement between annotators, or a measure of crowdsourced validity?

- Authors only report that most question templates were unique

Would newer models work better (i.e. BERT)?

Could automated question generation help?

Templates reduce the cost of generating questions for each relation, but it still costs annotator time and annotator knowledge

Pampari et al. 2018: algorithm for automated, large-scale (medical) QA dataset generation

- Extremely repetitive question sets (many questions had minimal variation)
- Personal experiments found repetitive questions unhelpful

Automated question generation could increase size of supervised data

Pampari, Anusri, et al. "EmrQA: A Large Corpus for Question Answering on Electronic Medical Records." *ArXiv:1809.00732 [Cs]*, Sept. 2018. *arXiv.org*, http://arxiv.org/abs/1809.00732.

To summarize,

Levy and colleagues contributed towards Zero-Shot Relation Extraction.

They framed Relation Extraction as a Reading Comprehension problem

Under this paradigm, a RE model performed well compared to other RE models in a zero-shot learning task

RE as QA still has short comings

- Expensive datasets
- Inferior compared to humans

Questions?

In positive examples, different types of cues can be seen

Relation: Solution by matching relation to question
Type: Solution relies on answer type
Verbatim: question appears in text
Global: phrasing in text differs from that in question
Specific: phrasing in text differs uniquely for a question

Verbatim	Relation	András Dombai **plays for** what team? András Dombai... ...currently **plays** as a goalkeeper **for** *FC Tatabánya*.
	Type	Which **airport** is most closely associated with Royal Jordanian? Royal Jordanian Airlines... ...from its main base at *Queen Alia International* ***Airport***...
Global	Relation	Who was responsible for **directing** Les petites fugues? Les petites fugues is a 1979 Swiss comedy film **directed by** *Yves Yersin*.
	Type	**When** was The Snow Hawk released? The Snow Hawk is a ***1925*** film...
Specific	Relation	Who **started** Fürstenberg China? The Fürstenberg China Factory **was founded**... ...**by** *Johann Georg von Langen*...
	Type	What **voice type** does Étienne Lainez have? Étienne Lainez... ...was a French operatic ***tenor***...

Figure 5: The different types of discriminating cues we observed among positive examples.

	Relation	Type
Verbatim	12%	5%
Global	8%	25%
Specific	22%	28%

Table 4: The distribution of cues by type, based on a sample of 60.

	Relation	Type
Verbatim	*43%*	*33%*
Global	*60%*	73%
Specific	46%	18%

Table 5: Our method's accuracy on subsets of examples pertaining to different cue types. Results in *italics* are based on a sample of less than 10.