Contrastive Triple Extraction with Generative Transformer

Hongbin Ye, Ningyu Zhang, Shumin Deng, Mosha Chen, Chuanqi Tan,

Fei Huang, Huajun Chen

yehb@zju.edu.cn

2020.12.19
Outline

• Motivation
• Related Work
• Approach
• Experiments
• Conclusions
Motivation

Triple Extraction

| Input                                                                 | The *United States* President *Trump* was raised in the borough of *Queens* in *New York City*, and lived there until age 13. |
|----------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| Output                                                               | Trump→president→of→United→States→[S2S_SEQ]→Trump→born→in→Queens→[S2S_SEQ]→Trump→live→in→Queens                       |
| Gold                                                                 | (Trump, president_of, United States)                                                                                  |
|                                                                      | (Trump, born_in, Queens)                                                                                              |
|                                                                      | (Trump, live_in, Queens)                                                                                              |
| Negative                                                             | (Trump, president_of, Queens)                                                                                         |
|                                                                      | (Trump, born_in, 13)                                                                                                   |
|                                                                      | (Trump, live_in, 13)                                                                                                   |
Extractive Models

- **Tagging(2017)** is an end-to-end method that uses a novel tagging scheme.
- **HRL(2019)** addresses relation extractions by regarding related entities as the arguments of the relation via hierarchical reinforcement learning.
- **MrMep(2019)** is an approach that utilizes triplet attention to exploit connections between relations and their corresponding entity pairs.
- **CasRel(2020)** is an approach that models relations as functions, which map subjects to objects in a sentence.
Related Work

Generative Models

• **CopyRE(2018)** is a Seq2Seq learning framework having a copy mechanism wherein multiple decoders are applied to generate triples to handle overlapping relations.

• **PNDec(2019)** provides two novel approaches using encoder-decoder architecture for triples having multiple tokens.

• **CopyMTL(2020)** proposes a multitask learning framework used to complete the entities.
Approach

Overlapping triples, Multiple tokens, Connection between multiple relationships
• End-to-end text sequence generative modeling

Capture long-term dependencies
• Transformer architecture

S2S architecture can generate faithful triples
• Contrastive learning mechanism
Approach

\[
\text{loss} = \text{loss}_{\text{generative}} + \alpha \text{loss}_{\text{contrastive}}
\]

[Diagram of the approach, showing the generative transformer and triplet contrastive learning.]
## Experiments

### Dataset

| Dataset     | NYT  | WebNLG | MIE   |
|-------------|------|--------|-------|
| Domain      | News | Web    | Medical |
| Relation    | 24   | 246    | 343   |
| Triplets    | 104,518 | 12,863 | 18,212 |
# Experiments

| Model                          | NYT  |  | WebNLG  |  |
|-------------------------------|------|---|---------|---|
|                               | P    | R | F     | P | R | F   |
| **Extractive**                |      |   |        |   |   |     |
| Tagging (Zheng et al. 2017b) | 61.5 | 41.4 | 49.5  | - | - | -   |
| HRL (Takanobu et al. 2019)   | 71.4 | 58.6 | 64.4  | 53.8 | 53.8 | 53.8 |
| MrMep (Chen et al. 2019)     | 77.9 | 76.6 | 77.1  | 69.4 | 77.0 | 73.0 |
| CasRel (Wei et al. 2020b)    | 89.7 | 89.5 | 89.6  | 93.4 | 90.1 | 91.8 |
| **Generative**                |      |   |        |   |   |     |
| CopyRE (Zeng et al. 2018b)   | 61.0 | 56.6 | 58.7  | 37.7 | 36.4 | 37.1 |
| PNDec (Nayak and Ng 2019)    | 80.6 | 77.3 | 78.9  | 38.1 | 36.9 | 37.5 |
| CopyMTL (Zeng, Zhang, and Liu 2020) | 75.7 | 68.7 | 72.0  | 58.0 | 54.9 | 56.4 |
| **Ours**                      |      |   |        |   |   |     |
| CGT (Random)                  | 90.8 | 77.7 | 83.7  | 87.6 | 70.5 | 78.1 |
| CGT (UniLM)                   | 94.7 | 84.2 | 89.1  | **92.9** | 75.6 | **83.4** |
| w/o contrastive               | 87.3 | 81.5 | 84.3  | 94.6 | 70.5 | 80.8 |

| Model        | P  | R  | F  |
|--------------|----|----|----|
| Bi-LSTM      | 53.13 | 49.46 | 50.69 |
| MIE-multi    | 70.24 | 64.96 | 66.40 |
| CGT(random)   | 70.75 | 66.96 | 68.80 |
| CGT(UniLM)    | **80.53** | **78.83** | **79.42** |
Experiments

Whether CGT can capture long-term dependence or not?
Case Study

Instance

instance #1 Batchoy is originates from the Philippines and served as a soup. Its main ingredients are noodles, pork organs, vegetables, chicken, shrimp and beef.
generated triple: \(\langle Batchoy, location, Philippines \rangle\)
ground truth: \(\langle Batchoy, country, Philippines \rangle\)

instance #2 Alan Shepard was a crew member of NASA operated Apollo 14 who died in California which is represented by Dianne Feinstein.
generated triple: \(\langle Shepard, deathPlace, California \rangle\)
ground truth: \(\langle Allan Shepard, deathPlace, California \rangle\)

instance #3 Saranac Lake, which is served by Adirondack Regional Airport, is part of Harrietstown, Essex County, New York, US.
generated triple: \(\langle Airport, cityServed, New York \rangle\)
ground truth: \(\langle Airport, cityServed, York \rangle\)
Conclusions

• We revisit triple extraction as a sequence generation task and introduce a novel CGT model. In light of the added extraction capability, CGT requires no additional parameters beyond those found in the pre-trained language model.

• We evaluate CGT on three benchmark datasets. Our model empirically outperforms other substantially strong baseline models. We also demonstrate that CGT is better than existing triple extraction approaches at capturing long-term dependencies, thus, achieving better performance with long sentences.

• In the future, we will utilize stronger transformer architectures, such as Longformer to generate relational knowledge from documents. We will also delve into injection ontology knowledge using condition generation methods. It will also be useful to apply our approach to other scenarios, such as event extractions.
THANKS

2020.12.19