Memory Enhanced Document-level Joint Entity and Relation Extraction

Witold Kościukiewicz[1,2] **Mateusz Wójcik**[1,2] **Tomasz Kajdanowicz**[2] **Adam Gonczarek**[1]

[1] Alphamoon Ltd., Grabarska 1, 50-072 Wrocław [2]Wroclaw University of Science and Technology

Contact: witold.kosciukiewicz@alphamoon.ai

Problem

Joint entity and relation extraction can be treated as a text-to-graph, task composed of **four NLP subtasks**:

- **mention detection**
- **entity classification**
- **coreference resolution**
- **relation classification**

We propose a **deep learning method** based on **memory**-like modules enhancing input representation, resulting in better multi-task learning process. Solving this problem on the **document-level** demands additional, intra-sentence inference based on coreferring pieces of information in a **different part of the document**. The result is a graph of spans from the text that represents mentions, grouped in coreference clusters, typed, and connected to each other by relations.

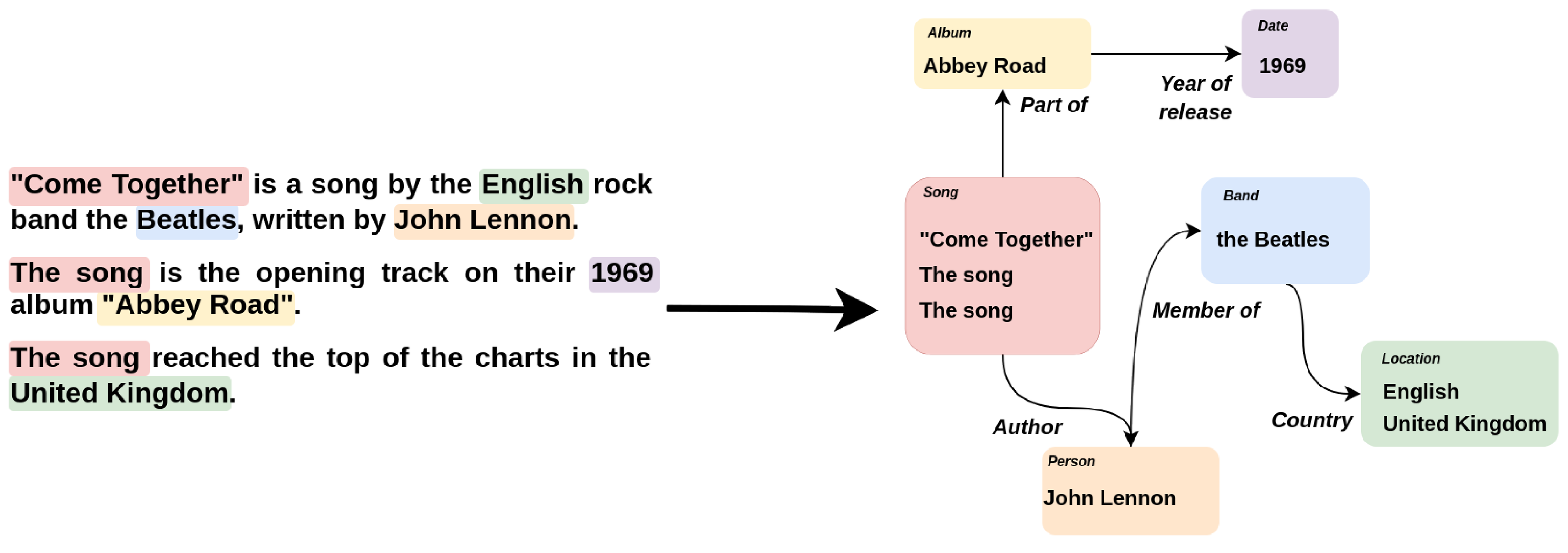

Figure 1: Document-level Joint entity and relation extraction as text to graph problem.

Model

We used multi-task learning architecture for joint entity and relation extraction proposed in [1], which processes tasks in a pipeline - one after another.

We improve and extend it by:

- introducing custom and writable memory matrix;
- backward passing of knowledge to earlier tasks from the latter by reading memory to alter the representations of tokens and mentions;

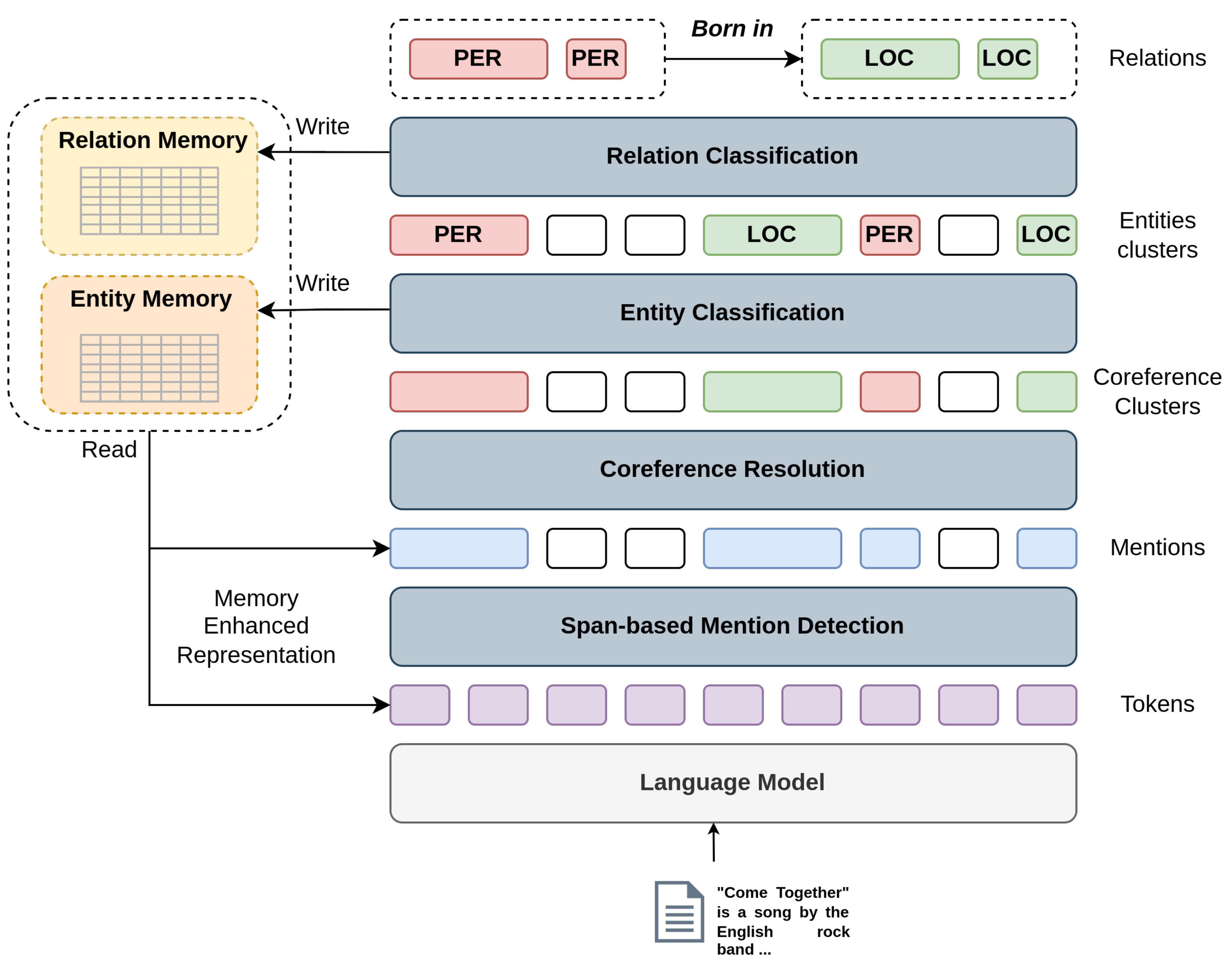

Figure 2: Proposed model architecture.

Memory Enhanced Representation

Based on writable Memory $\mathbf{M}$ and learnable memory reading weights $\mathbf{W}$ we propose two methods of altering input $\mathbf{H}$ using memory as in [2].

***Normal* memory read:**

$$\mathrm{Read}_{\mathrm{norm}}(\mathbf{H}, \mathbf{M}) = \mathrm{softmax}(\mathbf{H}\mathbf{W}\mathbf{M}^{\top})\mathbf{M}$$

***Inverse* memory read:**

$$\mathrm{Read}_{\mathrm{inv}}(\mathbf{H}, \mathbf{M}) = \sum_{i=1}^{|\mathbf{M}|} \mathrm{softmax}(\mathbf{M}_i\mathbf{W}\mathbf{H}^{\top})\mathbf{M}$$

Pooling methods:

- *mean* pooling for ***inverse*** read;
- concatenation for ***normal*** read;

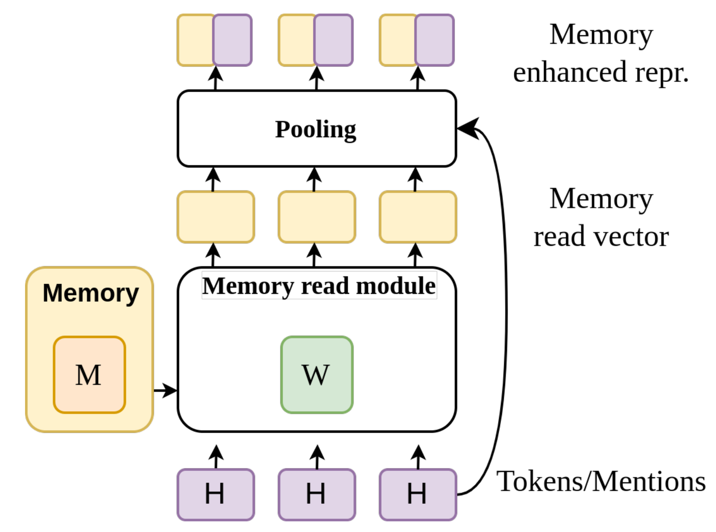

Figure 3: Memory read module used to enhance input representation

Memory Write Operation

Memory stores learnable representations used to classify entities and relations instances using bilinear similarity S between input $\mathbf{E}$ and memory $\mathbf{M}$.

Class distribution for entity e_i:

$$\mathrm{p}(\mathrm{e_i}) = \mathrm{softmax}(S(\mathbf{E}_{\mathbf{e_i}}, \mathbf{M}^{\mathbf{E}}))$$

Existance of relation $r_{i,j}$ between entities i and j:

$$\mathrm{p}(\mathrm{r_{i,j}}) = \mathrm{sigmoid}(S(\mathbf{E}_{\mathbf{r_{i,j}}}, \mathbf{M}^{\mathbf{R}}))$$

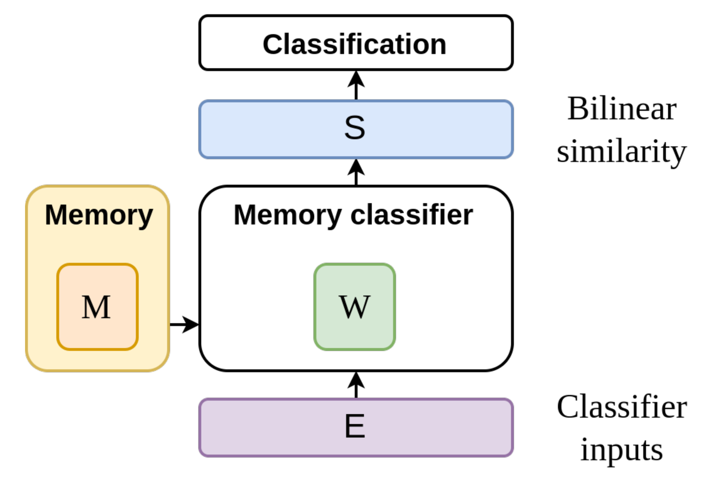

Figure 4: Memory write module using bilinear similarity classifier input $\mathbf{E}$ and memory $\mathbf{M}$ content

Experimental setup

We conducted several experiments trying different architecture parameters such as memory size and memory warmup to evaluate quality of proposed method. We trained and evaluated our model on **DocRED**[3]: dataset (*Human-annotated*) following the same train-test split as in [1] .

Dataset:

DocRED (*Human-annotated*) dataset:

- 5053 documents;
- 5 entity types;
- 96 relation types;

Experiments:

- *Adam* optimizer;
- *5e-5* learning rate with 0.1 *Linear Warmup*;
- batch size: 2;
- repeated for 5 different random seeds;

Experiments results

To compare proposed method to reproduced results from [1] we used F1-score (micro) on each of 4 subtasks. Presented results are based on 5 separate runs with standard deviation provided.

Model	Rel. F1-score	Ent. F1-score	Coref. F1-score	Men. F1-score
JEREX (*GRC*) [1]	38.65 ±0.13	79.66 ±0.26	82.35 ±0.25	92.54 ±0.13
ours (*GRC*)	**38.91** ±0.15	**79.72** ±0.06	**82.43** ±0.06	**92.63** ±0.12
JEREX (*MRC*) [1]	**39.93** ±0.47	79.68 ±0.20	82.41 ±0.17	92.54 ±0.12
ours (*MRC*)	39.89 ±0.27	**79.71** ±0.31	**82.42** ±0.28	**92.77** ±0.17

References

[1] Markus Eberts and Adrian Ulges. An end-to-end model for entity-level relation extraction using multi-instance learning. *arXiv preprint arXiv:2102.05980*, 2021.

[2] Yongliang Shen, Xinyin Ma, Yechun Tang, and Weiming Lu. A trigger-sense memory flow framework for joint entity and relation extraction. In *Proceedings of the web conference 2021*, pages 1704–1715, 2021.

[3] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. Docred: A large-scale document-level relation extraction dataset. *arXiv preprint arXiv:1906.06127*, 2019.