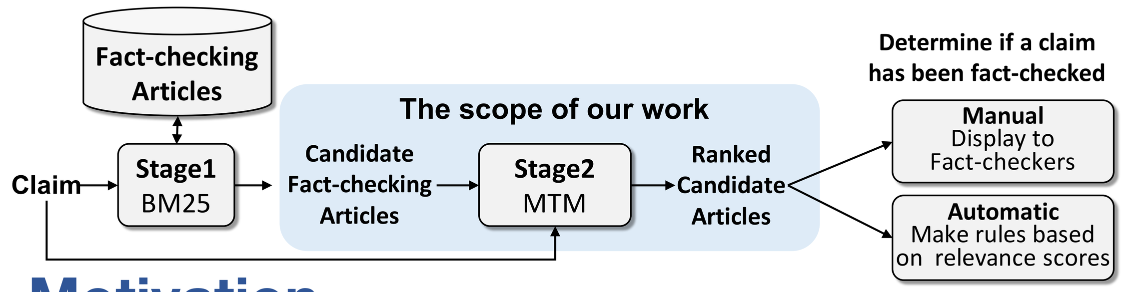
Article Reranking by Memory-Enhanced Key Sentence Matching for Detecting Previously Fact-Checked Claims

ACL-IJCNLP 2021

Qiang Sheng[1,2], Juan Cao[1,2], Xueyao Zhang[1,2], Xirong Li[3], Lei Zhong[1,2]
[1]Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences [3]Key Lab of Data Engineering and Knowledge Engineering, Renmin University of China

Introduction

- **Background:** False claims that have been **previously fact-checked** can still spread on social media.
- **Task:** Given a claim, retrieval against the fact-checking article (FC-article) collections to find the corresponding fact-checking articles, if any.
- **Formulation:** Typically formulated as a two-stage retrieval-based workflow.
- We focus on matching the claim with key sentences in FC-articles for better **reranking**.



Motivation

Existing works ignore the following characteristics of FC-articles:

- Claims are often *quoted* in key sentences. (**Lexical Information**, e.g., S2)
- Templates to introduce or debunk claims are common across articles. (**Pattern Information**, e.g., S2 and S3)

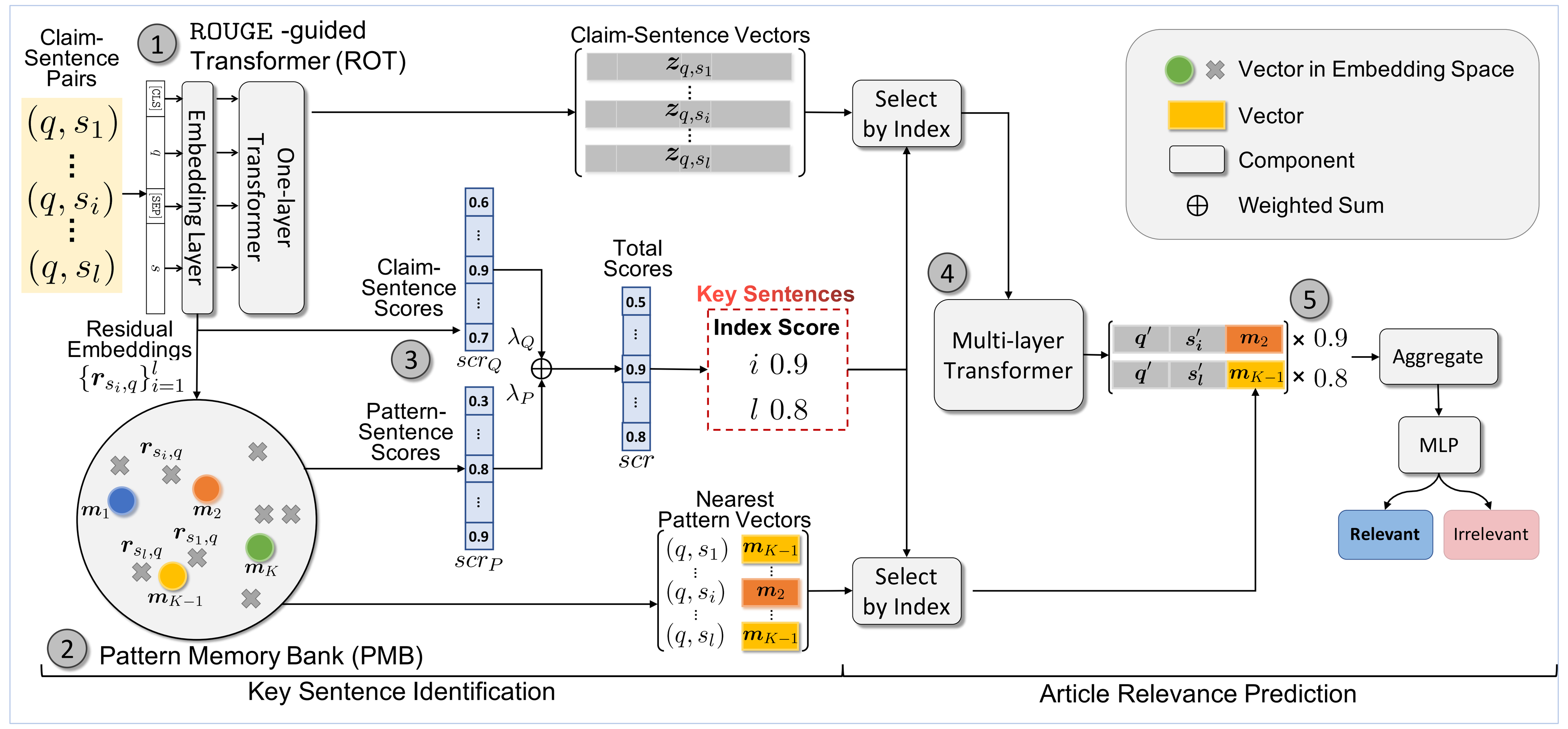
Claim Hot lemonade can kill cancer cells without hurting normal cells.

Sentences in Candidate Fact-checking Articles

Sentence	Relevant?	Contains Quotation?	Contains Fact-checking Patterns?
S1. Lemon is not so-called acid food, and drinking lemonade does not lead to cancer. (From Article 1)	No, but on a similar topic	No	No
S2. The rumor saying hot lemonade can kill cancer cells has spread over years. (From Article 2)	Yes	Yes (Underlined)	Yes (In boldface)
S3. It is just a groundless inference that lemon has a curative effect of cancer. (From Article 3)	Yes	No	Yes (In boldface)

Our Method

MTM (Memory-Enhanced Transformers for Matching)



Input: The given claim *q* paired with each sentence *s* in the candidate FC-article *d*.

Step 1: Key Sentence Identification

① **ROUGE-guided Transformer (ROT):** The claim-sentence (*q*-*s*) pairs are fed into ROT to obtain claim-sentence vectors and scores. The ROT is a part of pretrained BERT that was finetuned with ROUGE scores as supervision, which considers both semantic and ***lexical information*** in *q*-*s* pairs.

② **Pattern Memory Bank (PMB):** The PMB stores the *pattern vectors*, which are initialized with the clustering results of the embedding differences in all candidate *q*-*s* pairs. The distance between the residual embedding (subtracting *q* from *s*) and the nearest pattern vector indicates how possible *s* contains ***pattern information***.

③ **Key Sentence Selection:** The claim-sentence scores (from ROT) and the pattern-sentence scores (from PMB) are aggregated to rank each sentence in *d*. Top *k* sentences are regarded as *d*'s proxy.

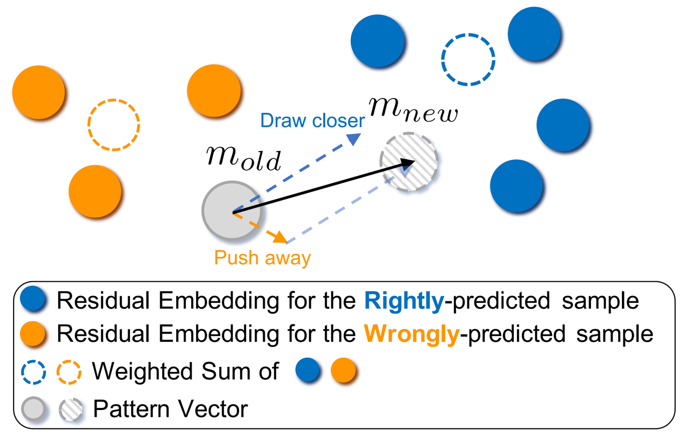
Step 2: Article Relevance Prediction

④ **Sentence Representation:** The selected top *k* *q*-*s* vectors are fed into a multi-layer transformer to obtain richer representation.

⑤ **Weighted Memory-aware Aggregation:** The nearest pattern vectors w.r.t. top *k* pairs are concatenated to the sentence representations. Then the vectors are aggregated with scores in ③ as weights to obtain *q*-*d* vector for the final prediction of whether *d* fact-checks *q*.

Memorized Pattern Vectors Update during Training

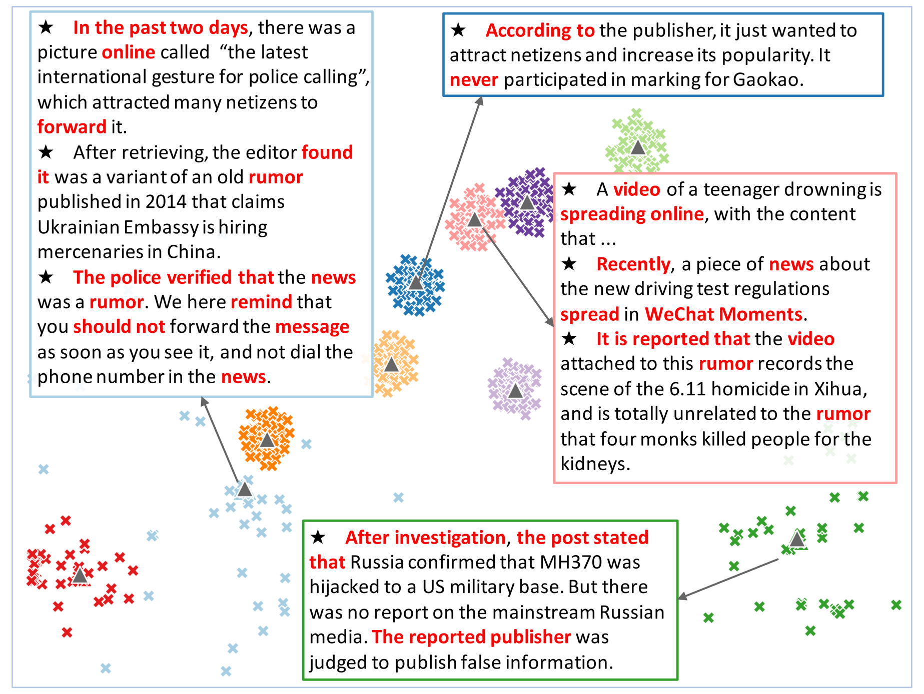
- An epoch-wise update.
- For each pattern vector *m*, we collect all residual embeddings whose nearest pattern vector is *m*.
- We update *m* to make it closer to the rightly-predicted samples and far away from those wrongly-predicted ones.



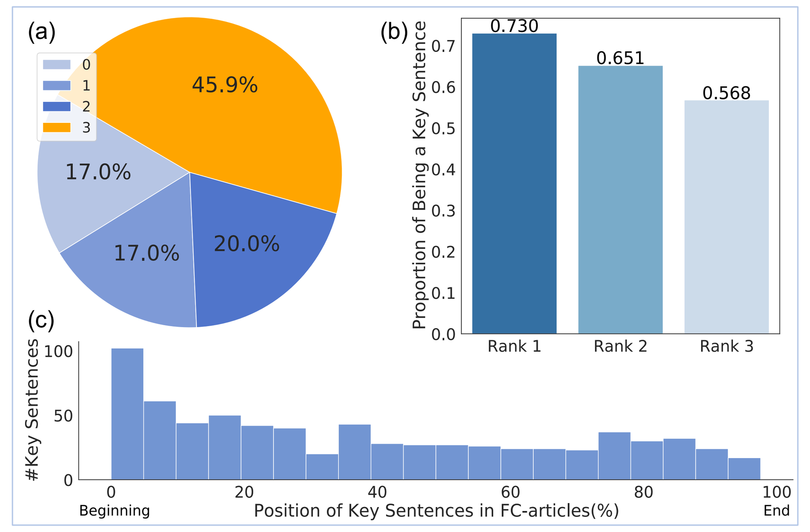
Evaluation

Table 2: Performance of baselines and MTM. Best results are in **boldface**.

Method	Selecting Sentences?	Weibo						Twitter					
		MRR	MAP@			HIT@		MRR	MAP@			HIT@	
			1	3	5	3	5		1	3	5	3	5
BM25		0.709	0.355	0.496	0.546	0.741	0.760	0.522	0.460	0.489	0.568	0.527	0.568
BERT		0.834	0.492	0.649	0.693	0.850	0.863	0.895	0.875	0.890	0.890	0.908	0.909
DuoBERT		0.885	0.541	0.713	0.756	0.886	0.887	0.923	**0.921**	0.922	0.922	0.923	0.923
BERT(Transfer)	✓	0.714	0.361	0.504	0.553	0.742	0.764	0.642	0.567	0.612	0.623	0.668	0.719
Sentence-BERT	✓	0.750	0.404	0.543	0.589	0.810	0.861	0.794	0.701	0.775	0.785	0.864	0.905
RankSVM	✓	0.809	0.408	0.607	0.661	0.887	0.917	0.846	0.778	0.832	0.840	0.898	0.930
CTM		0.856	0.356	0.481	0.525	0.894	0.935	0.926	0.889	0.919	0.922	0.952	0.964
MTM	✓	**0.902**	**0.542**	**0.741**	**0.798**	**0.934**	**0.951**	**0.931**	0.899	**0.926**	**0.928**	**0.957**	**0.967**



↑ MTM outperforms all compared methods on the two datasets (the exception is only the MAP@1 on Twitter).
↗ Visualization of residual embeddings around pattern vectors shows the ability of MTM to mine the fact-checking patterns.
→ Human evaluation on 370 samples show that MTM can find at least one key sentence in 83% of FC-articles (a) and 73% are at Rank 1 (b), even though the positional distribution of key sentences are scattered throughout the articles (c).



Conclusion

- **Method:** We propose MTM to select from FC-articles key sentences that introduce or debunk claims, and exploit the selected sentences for estimating the relevance of the FC-articles w.r.t. a given claim.
- **Evaluation**: Experiments show that MTM outperforms existing methods. Further human evaluation and case studies prove that our model can find key sentences, which can be regarded as explanations.
- **Data:** We built the first Chinese dataset for fact-checked claim detection with fact-checking articles from diverse sources.

Our Project

https://tinyurl.com/sbe3s259