KPatch: Knowledge Patch to Pre-trained Language Model for Zero-Shot Stance Detection on Social Media

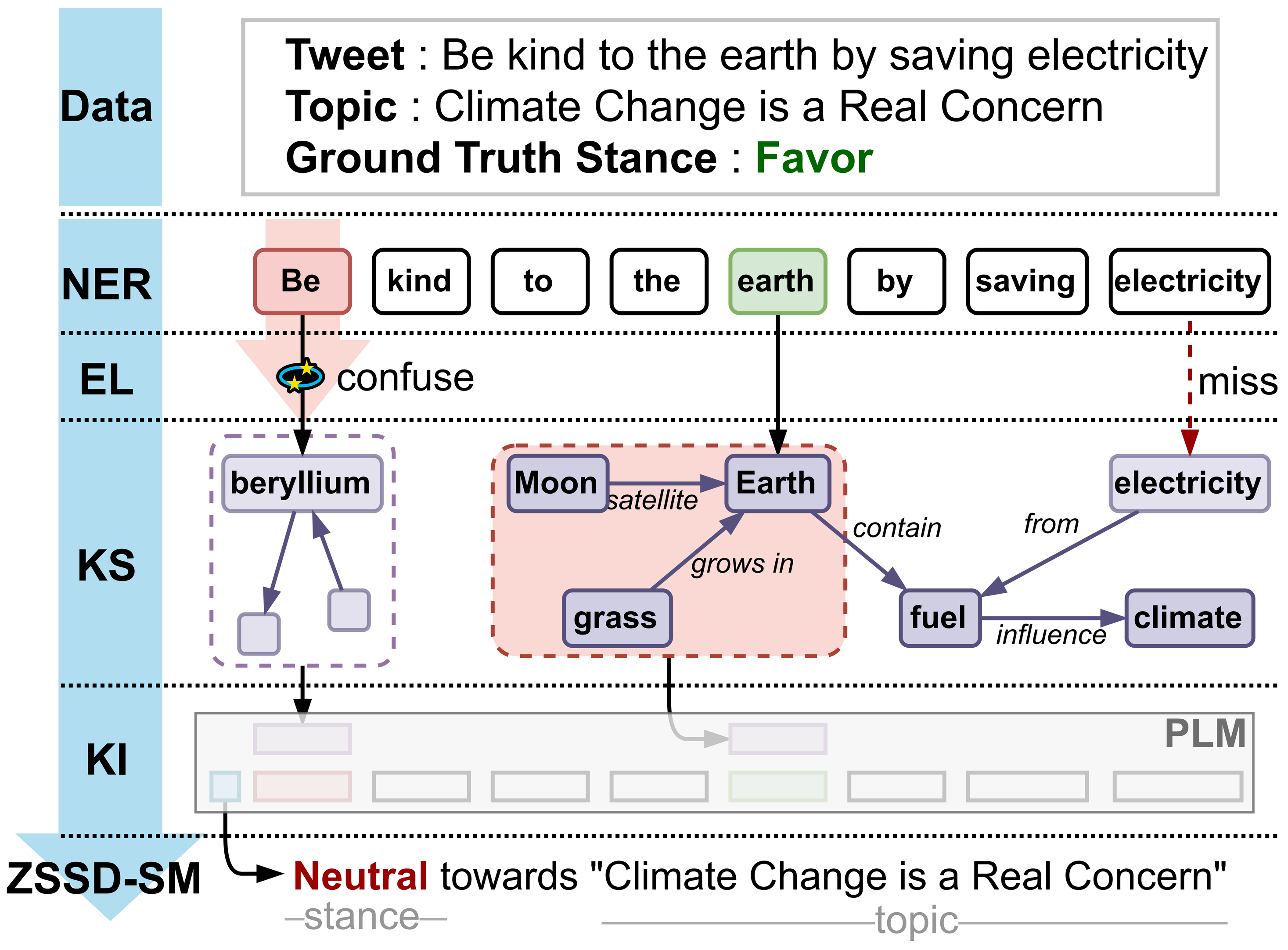
Shuohao Lin[1,3], Wei Chen[1,3], Yunpeng Gao[2], Zhishu Jiang[1,3], Mengqi Liao[1,3], Zhiyu Zhang[1,3], Shuyuan Zhao[1,3], Huaiyu Wan[1,3,*]

[1] Beijing Jiaotong University [2] Zhipu AI [3] Beijing Key Laboratory of Traffic Data Analysis and Mining

Introduction & Challenges

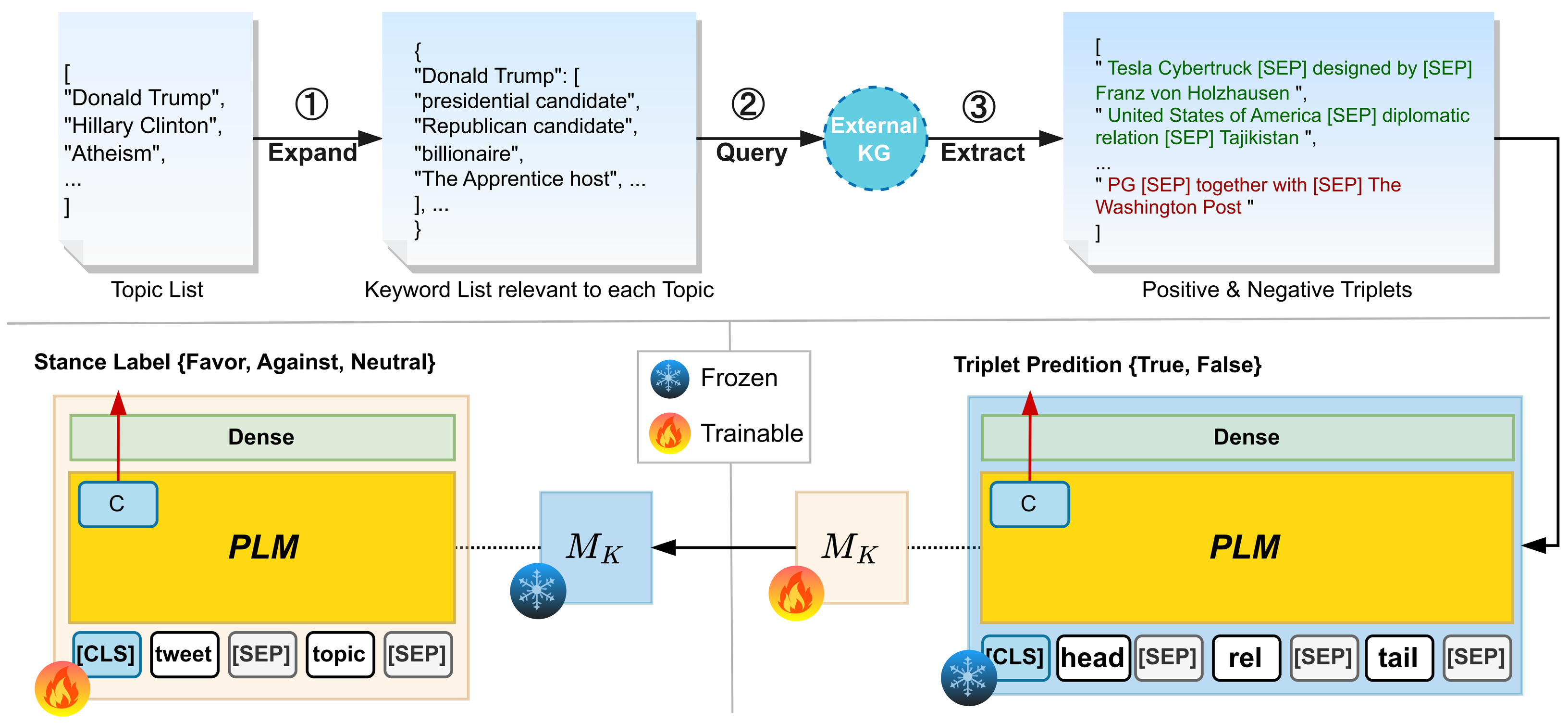
Zero-shot stance detection on social media (ZSSD-SM) aims for determining the attitude or standpoint expressed in tweets towards an unseen target.

Traditional domain adaptation methods are difficult to transfer between two domains with knowledge gaps, while the knowledge injection (KI) methods commonly used to alleviate this problem are affected by the lengthy preparatory work, which can lead to **error accumulation** and **irrelevant knowledge injection**.



The challenges of traditional KI methods in handling ZSSD-SM task.

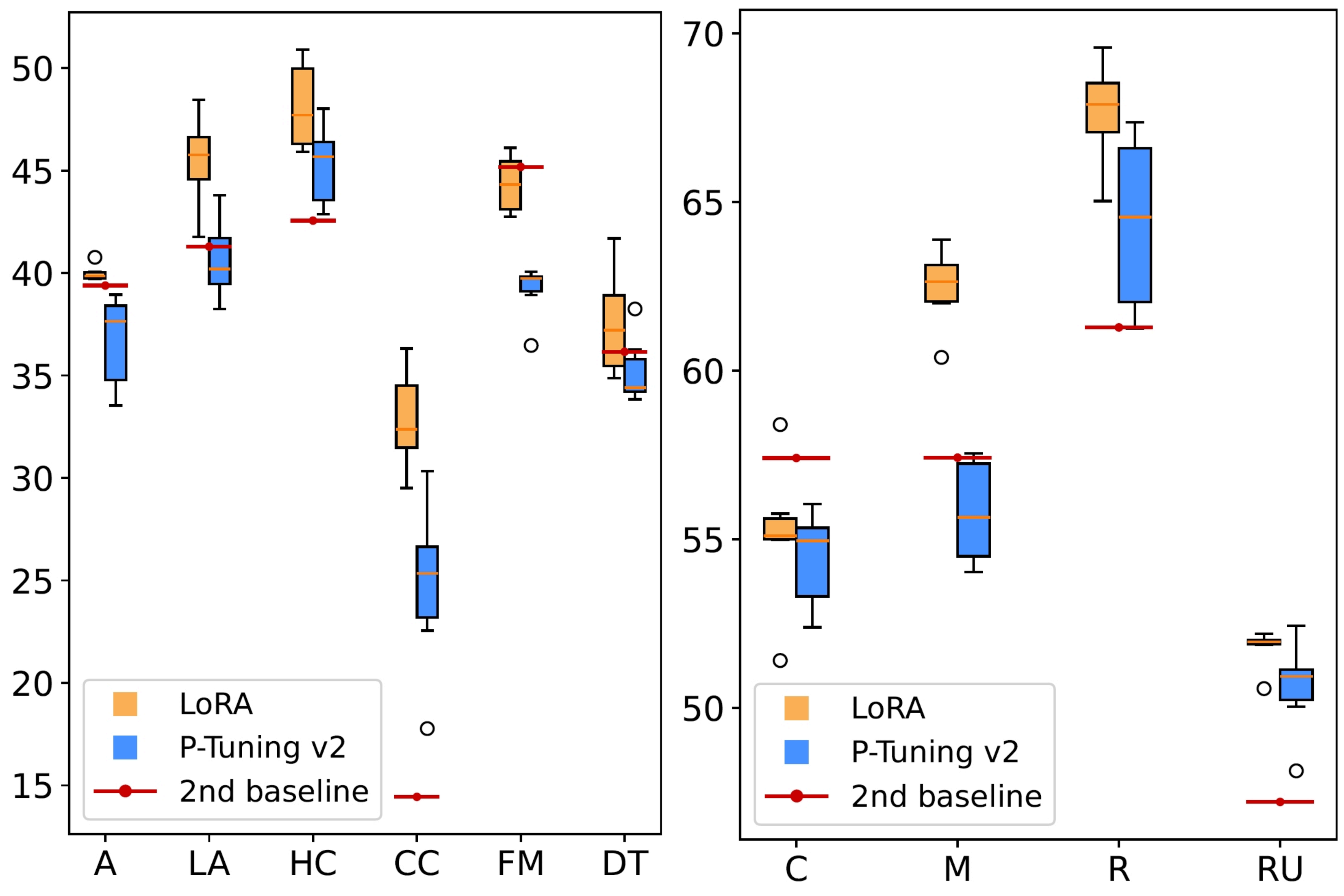
The Proposed KPatch Framework



The framwork of KPatch.

1. **Knowledge Searching**: Extract triplets from subgraph as positive cases, and randomly replace the head or tail entity to form negative cases.
2. **Knowledge Compression Stage**: Train the M_K in Triplet Denoising Task to compress the knowledge into the M_K (freeze PLM and the dense layer).
3. **Task Guidance Stage**: Freeze the M_K from the Comparison Stage, and then fine-tune the PLM and the dense layer on the stance detection task.

Experiments



Sem16 MiDe22-EN

Performance fluctuations under different PEFT and random seeds.

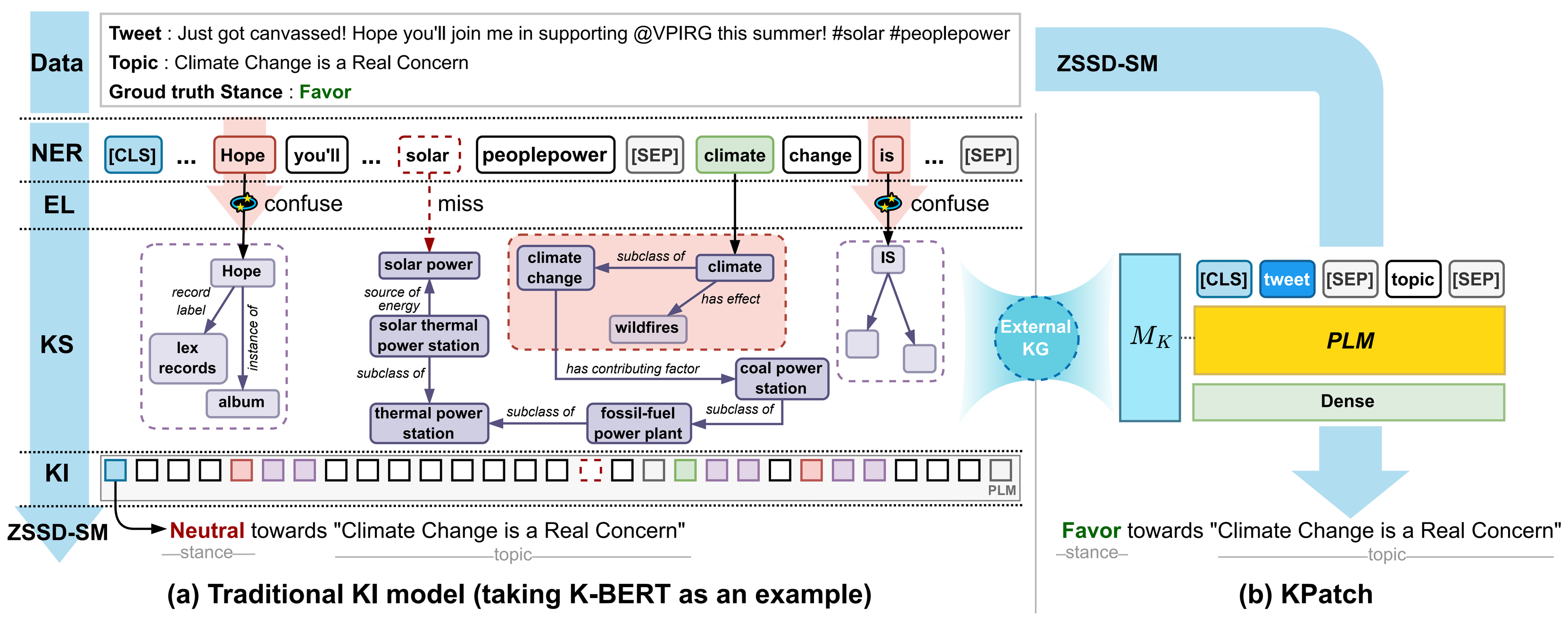
Performance gains of different KI methods compared to backbone PLM.

Model	Sem16 (ΔF_{avg}%) A	LA	HC	CC	FM	DT	MiDe22-EN (ΔF_{macro}%) C	M	R	RU
K-BERT	0.22	-2.83	-1.32	-8.23	**1.56**	-1.38	**5.76**	1.09	0.12	3.35
KP(LoRA)	**9.12**	6.36	**7.19**	**17.48**	0.87	20.84	-2.35	8.12	**8.22**	**7.70**
KP(PTV2)	7.87	**7.50**	3.29	13.55	1.11	**21.51**	0.73	**10.41**	7.74	5.02

Ablation experiments.

Model	Sem16 (F_{avg}%) A	LA	HC	CC	FM	DT	MiDe22-EN (F_{macro}%) C	M	R	RU
KP(LoRA)	**39.86**	43.80	**49.75**	**31.92**	43.92	41.14	55.06	60.55	**69.50**	**54.92**
w/o Expand	34.44	**46.26**	48.44	22.18	42.52	35.68	54.08	57.31	67.85	52.72
KP(PTV2)	38.61	44.94	45.85	27.99	**44.16**	**41.81**	**58.14**	**62.84**	69.02	52.24
w/o Expand	36.32	39.00	43.98	18.03	37.19	33.66	55.48	55.60	66.61	48.55
w/o Compress	39.83	45.36	48.69	29.84	39.32	36.04	48.74	53.70	59.60	48.42

Case Study & Conclusion



Comparison of the prediction process between K-BERT and KPatch.

- **KPatch (b)** hands over these manually designed steps to the PLM for processing and implements knowledge injection through the hidden space, while K-BERT (a) requires the lengthy preparatory work before the KI stage.
- **KPatch** skips the preparatory work of traditional knowledge injection pipeline, which guides the PLM to adaptively select the most suitable knowledge from the external matrix through its latent modeling ability for ZSSD-SM task.