Effective and Unsupervised Social Event Detection via RAG

Qitong Liu
Beihang University
Beijing, China
liuqt@buaa.edu.cn

Hao Peng*
Beihang University
Beijing, China
penghao@buaa.edu.cn

Abstract

With the growing scale of social media, social event detection has attracted increasing attention. Graph neural networks (GNNs) and transformer-based pre-trained language models (PLMs) have become mainstream approaches in this area. However, existing methods still face two major challenges. First, the sheer volume of social media messages leads to resource-intensive learning. Second, the fragmentation of social media messages often impedes the model's ability to capture a comprehensive view of the events. To address these challenges, we propose a foundation model for unsupervised Social Event Detection, namely RagSED. Specifically, RagSED introduces a representativeness and diversity-driven sampling strategy to extract key messages from massive social streams, significantly reducing noise and computational overhead. It further establishes a novel paradigm based on Retrieval-Augmented-Generation (RAG) that enhances PLMs in detecting events while simultaneously constructing and maintaining an evolving event knowledge base. Extensive experiments on a public dataset demonstrate the superiority of RagSED in open-world social event detection.

CCS Concepts

• **Information systems** → **Social networks**.

Keywords

Social event detection, Retrieval-augmented generation, Large language model

1 Introduction

Social media platforms have become a vital source of real-time signals for emerging events, drawing increasing attention to the task of event analysis. The task of Social Event Detection (SED) focuses on automatically identifying and clustering social messages that pertain to the same real-world event [1].

However, existing approaches face several critical challenges in the context of open-world, unsupervised social event modeling. **First**, the massive volume and informal nature of social messages lead to resource-intensive computation and high noise levels, affecting detection precision. For popular events, there often exist large numbers of semantically similar or even completely identical messages. [1, 2, 8] use all messages for event detection, resulting in a serious waste of resources. Furthermore, users often post messages randomly, and a large number of messages lack attributes, which leads to the problems of missing edges and noisy edges in methods based on GNN. Although [12] employs message anchors to reduce computational overhead, it still relies on building a global graph using all messages, making it difficult to eliminate noisy edges. **Second**, the fragmented nature of social messages often prevents models from capturing the complete picture of events, resulting in shallow representations or misclustering. Existing methods rely solely on message-level embeddings during the clustering process, without incorporating any global semantic guidance from already occurred events. Ignoring these global signals results in the model's inability to align new messages with the broader narrative, particularly in cases involving ambiguous phrasing or topic drift [4].

To tackle these challenges, we propose a novel foundation model for unsupervised Social Event Detection, namely RagSED. To address the aforementioned challenges, our proposed framework consists of two parts: key message sampling and RAG-based event detection. **First**, to mitigate the computational inefficiency caused by highly semantically similar messages, RagSED designs a sampling strategy that balances representativeness and diversity to extract key messages from the social message stream. By using only these key messages, rather than the entire messages for downstream event detection, RagSED meets the requirements of efficient and real-time detection in open-world settings. **Second**, leveraging the powerful contextual reasoning capabilities of PLMs, RagSED introduces a novel RAG-based event detection paradigm. In contrast to prior methods that follow a "representation-then-cluster" approach, RagSED constructs and maintains a structured knowledge base of detected events and employs RAG [5] to inject global semantic signals into PLMs during detection. Furthermore, since RagSED relies solely on the semantic content of messages, it avoids the issues of missing or noisy edges mentioned above.

In summary, the contributions of this paper are as follows:

• We propose a novel framework for streaming and unsupervised social event detection, termed RagSED, that integrates RAG and effectively addresses two key challenges in open-world social event analysis.

• We propose a novel paradigm for event detection by leveraging knowledge bases and RAG. This paradigm fully exploits the powerful contextual understanding capabilities of PLMs. By designing a key message sampling strategy, our method significantly reduces computational costs while maintaining detection effectiveness.

• Extensive experiments on a benchmark dataset demonstrate that our proposed RagSED consistently outperforms strong baselines in detection accuracy.

2 Methodology

In this section, we systematically describe the proposed RagSED. Section 2.1 presents a strategy for sample key messages. Section 2.2 introduces the event detection paradigm of RagSED based on RAG-enhanced PLMs.

2.1 Key Message Sampling (KMS)

Given a message block M_t in a high-throughput social stream, our goal is to select a set of key messages from M_t that preserves both

*Corresponding author.

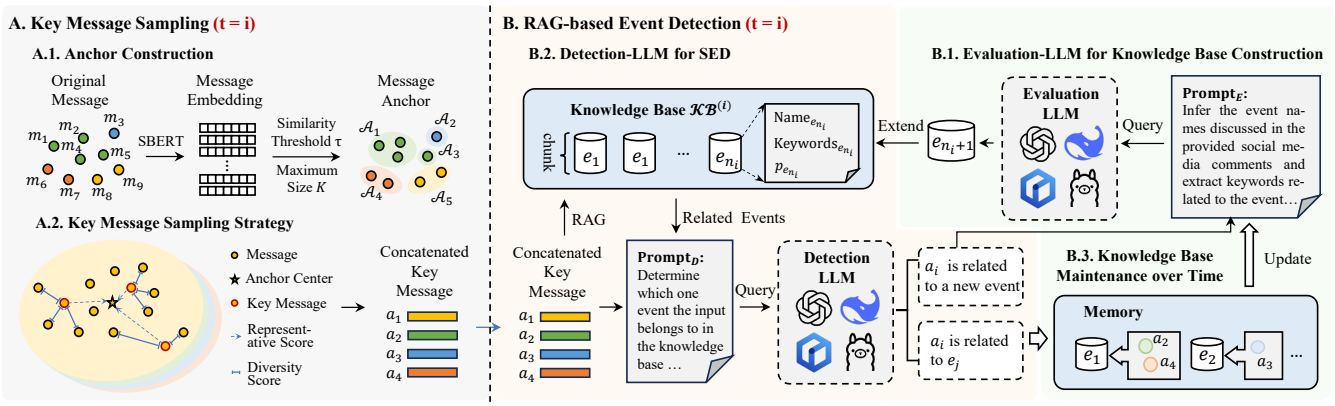

Figure 1: The proposed RagSED framework.

the representativeness of each message anchor and the diversity within each anchor, as shown in Figure 1(A). This lowers computation and maintains detection quality for downstream event detection.

2.1.1 Anchor Construction. This step aims to aggregate messages that are highly semantically similar or even exactly identical into anchors. We first encode each message m_i into a z_i. Then, we compute the cosine similarity s_{ij} between two messages. Messages are assigned to the same anchor $\mathcal{A}_k$ if and only if

$$s_{ij} \geq \tau, \quad \forall m_i, m_j \in \mathcal{A}_k, \tag{1}$$

where $\tau \in (0, 1)$ is a predefined similarity threshold. To avoid oversize anchors for popular events, we use the maximum size K that limits the number of messages in each anchor. If adding a new message would cause $|\mathcal{A}_k| > K$, a new anchor is created. All messages can be represented as an anchor collection $\mathbb{A} = \{\mathcal{A}_1, \mathcal{A}_2, \dots\}$.

2.1.2 Key Message Sampling Strategy. To select key messages from each anchor that are central and non-redundant within each anchor, we define two scoring criteria: representativeness and diversity.

Representativeness Score. The representativeness score measures how well a message reflects the central semantics of its anchor. Formally, for a message m_i in anchor $\mathcal{A}_k$, its representativeness score is defined as

$$\mathrm{Rep}(m_i) = \frac{z_i^\top c_k}{\|z_i\|_2 \, \|c_k\|_2}, \quad c_k = \frac{1}{n} \sum_{j=1}^{n} z_j, \tag{2}$$

where c_k is the center embedding of $\mathcal{A}_k$, and n is the message number of $\mathcal{A}_k$. A high representativeness score indicates strong semantic alignment with the anchor center, thereby capturing the dominant event of the anchor.

Diversity Score. The diversity score measures the degree to which a message provides novel information from the same anchor. For a message m_i in anchor $\mathcal{A}_k$, its diversity score is defined as

$$\mathrm{Div}(m_i) = \frac{1}{n-1} \sum_{m_j \in \mathcal{A}_k, j \neq i} \frac{z_i^\top z_j}{\|z_i\|_2 \, \|z_j\|_2}, \tag{3}$$

where n is the message number of $\mathcal{A}_k$. A high diversity score implies ensuring coverage of different aspects of the event.

Combined Score. The final score balances the two criteria:

$$\mathrm{S}(m_i) = \lambda \cdot \mathrm{Rep}(m_i) + (1 - \lambda) \cdot \mathrm{Div}(m_i), \tag{4}$$

where $\lambda \in [0, 1]$ controls the trade-off between representativeness and diversity. At each anchor $\mathcal{A}_k$, we select the top p messages with the highest combined scores as key messages and concatenate them as a_k for subsequent event detection.

2.2 RAG-based Event Detection

In this section, we present our RAG-based event detection framework, as shown in Figure 1(B), which is designed to incorporate both the construction and maintenance of an event knowledge base (Sections 2.2.1 and 2.2.3) and the use of RAG to leverage the knowledge base as external global guidance for event detection (Section 2.2.2). To ensure computational efficiency in the social stream, we do not rely on all incoming messages. Instead, we employ the key messages sampled in Section 2.1. Take M_t as an example, the final input set for detection is thus represented as $A = \{a_1, a_2, \dots\}$, where each a_k serves as the detection unit in subsequent steps.

2.2.1 Evaluation-LLM for Knowledge Base Construction. To facilitate retrieval-augmented event detection, we create and dynamically expand a knowledge base during the detection process. The knowledge base stores structured information of previously detected events, including their names, salient keywords, and embeddings. Formally, the knowledge base for M_t is defined as:

$$\mathcal{KB}^{(t)} = \{e = (\mathrm{Name}_e, \ \mathrm{Keywords}_e, \ p_e)\}, \tag{5}$$

, where Name_e denotes the event name of e, $\mathrm{Keywords}_e = \{w_1, \ w_2, \dots\}$ is the keyword set describing the event e, and $p_e \in \mathbb{R}^d$ is the event embedding. The embedding is obtained by encoding both the event name and keywords. Each event e is stored as an independent chunk in the knowledge base, enabling efficient retrieval and dynamic expansion during subsequent detection.

The event information, i.e., Name_e and $\mathrm{Keywords}_e$, is extracted from key messages using an Evaluation-LLM guided by our designed Prompt_E. The Prompt_E contains (i) a task description ("infer the event names discussed in the provided social media comments and extract keywords related to the event"), (ii) specific execution rules, and (iii) output requirements specifying a structured JSON form.

When new messages a_k arrive and cannot be aligned with any existing event in the knowledge base, the Evaluation-LLM is again invoked to generate a new event chunk:

$$e_{\text{new}} = \text{LLM}_{\text{Prompt}_E}(a_k) = (\text{Name}_{\text{new}},\ \text{Keywords}_{\text{new}},\ p_{\text{new}}), \quad (6)$$

where a_k is the aggregated message from the new anchor. The new event e_{new} is then inserted into the knowledge base. Through this incremental process, the knowledge base gradually accumulates structured information about various events, providing powerful global semantic guidance for detection.

2.2.2 Detection-LLM for SED. To enhance the capability of LLMs in social event detection, we design a RAG-based paradigm. Given an incoming aggregated message a_k, RAG first queries the knowledge base $\mathcal{KB}^{(t)}$ to retrieve the most semantically relevant candidate events by event embedding. Retrieval of RAG is performed using cosine similarity. Formally, for each event e in $\mathcal{KB}^{(t)}$:

$$r(e \mid a_k) = \frac{\text{Enc}(a_k)^\top p_e}{\|\text{Enc}(a_k)\|_2 \cdot \|p_e\|_2}, \quad (7)$$

where $r(e \mid a_k)$ represents the degree of correlation between event e and message a_k in the knowledge base, p_e represents the embedding of e, and Enc is the embedding model used to obtain p_e. Sort the correlation of all events in the knowledge base from high to low, and the top-q events are selected as the retrieval events:

$$\mathcal{N}_{E(a_k)} = \underset{e \in \mathcal{KB}^{(t)}}{\arg\max}^{q} \{r(e \mid a_k) \mid r(e \mid a_k) \geq \gamma\}, \quad (8)$$

where $\mathcal{N}_{E(a_k)}$ represents the retrieval event set, and γ represents the similarity threshold of RAG.

The Detection LLM is then prompted with both the query message a_k and the retrieval event set $\mathcal{N}_{E(a_k)}$. Our designed Prompt_D consists of (i) a task description ("determine which one event the input belongs to in the knowledge base"), (ii) specific execution rules, (iii) output requirements specifying a structured JSON form, and (iv) the retrieval event set as global semantic guidance. The detection decision is formalized as:

$$y_{a_k} = \text{LLM}_{\text{Prompt}_D}(a_k,\ \mathcal{N}_{E(a_k)}) = \begin{cases} e \in \mathcal{N}_{E(a_k)} \\ \text{Others} \end{cases}, \quad (9)$$

where y_{a_k} is the event a_k discussed obtained by Detection-LLM based on Prompt_D and the knowledge base. It is worth noting that when the retrieval event set is empty or irrelevant, the Detection-LLM output is "Others". For any a_k predicted as "Others", we treat it as belonging to a previously unseen event e_{new}, i.e., $y_{a_k} = e_{\text{new}}$. In this case, a_k is forwarded to the Evaluation-LLM, which generates the event name and keywords of e_{new}, and the knowledge base is updated accordingly to enable Detection-LLM to identify subsequent messages related to e_{new}.

Finally, according to y_{a_k}, the event label of the original message is obtained as:

$$y_{m_i} = y_{a_k}, \quad \forall\, m_i \in \mathcal{A}_k, \quad (10)$$

where $\mathcal{A}_k$ is the corresponding anchor of a_k. Through this RAG-based event detection paradigm, our framework effectively incorporates structured event information as global semantic guidance, significantly enhancing the ability of LLMs to perform accurate and efficient SED.

2.2.3 Knowledge Base Maintenance over Time (KBM). To prevent the event information in the knowledge base from becoming outdated during long-term detection, we design a maintenance mechanism that periodically refreshes event information with newly messages. Specifically, for each event $e \in \mathcal{KB}^{(t)}$, we maintain a buffer $\mathcal{B}_e$ to record the set of messages that have been assigned to e by the detection process. When the size of the message in the buffer exceeds a predefined threshold θ, the buffered messages are concatenated and forwarded to the Evaluation-LLM. The LLM generates refreshed event keywords, and we then update the knowledge base. After the update, the buffer is cleared, and the system continues to collect messages for e. Once the buffer again reaches the threshold θ, the update process is repeated.

3 Experiments

We conduct experiments to validate the effectiveness of the proposed RagSED. In addition, we conduct an ablation study.

3.1 Experimental Setups

3.1.1 Evaluation Metrics. To assess the effectiveness of SED, we follow previous studies [2, 11] and use three clustering metrics to measure the consistency between the detected event clusters and the ground truth clusters: Normalized Mutual Information (NMI) [3], Adjusted Mutual Information (AMI) [9], and Adjusted Rand Index (ARI) [9].

3.1.2 Datasets. We conduct experiments on a public Twitter dataset: Event2012 (68,841 English messages, 503 events) [6]. Following processing in [2], the datasets are split into daily message blocks.

3.1.3 Baselines. To evaluate the SED performance of the proposed RagSED, we compare it with two GNN-based methods (**KPGNN** [1] and **QSGNN** [8]), three PLM-based methods (**SBERT** [7], **CP-Tuning** [10], and **PromptSED** [11]), and one structural entropy-based method (**HISEvent** [2]).

3.2 Main Results

The SED evaluation results of RagSED are reported in Tables 1, where both Detection-LLM and Evaluation-LLM are instantiated with deepseek-r1-32b. RagSED consistently outperforms all baselines across nearly all message blocks in terms of NMI, AMI, and ARI, achieving maximum gains of 0.22 in NMI, 0.24 in AMI, and 0.63 in ARI (e.g., block M_{15}). These improvements highlight the effectiveness of global event guidance in forming more accurate event clusters. Importantly, RagSED maintains stable promotion in both dense message blocks (e.g., M_1, M_7, M_{12}) and sparse message blocks (e.g., M_{16}, M_{20}), demonstrating robustness to varying data distributions. In practical deployment, selecting larger LLMs can further enhance performance.

3.3 Ablation Study

We conduct ablation studies to further demonstrate the effectiveness of each module in RagSED. As shown in Table 2, removing the KMS, $\mathcal{KB}$, or KBM modules leads to varying degrees of performance degradation in SED. This is because the KMS module samples key messages instead of all messages, thereby improving the quality of LLM queries; the KB module provides event information as global

Table 1: Social event detection results on Events2012. * marks results acquired with the ground truth message labels. The best results are bolded, the second-best results are underlined, and the proposed method is marked in an orange background.

Blocks	M_1			M_2			M_3			M_4			M_5			M_6			M_7		
Metric	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI
KPGNN*	0.39	0.37	0.07	0.79	0.78	0.76	0.76	0.74	0.58	0.67	0.64	0.29	0.73	0.71	0.47	0.82	0.79	0.72	0.55	0.51	0.12
QSGNN*	0.43	0.41	0.07	0.81	0.80	0.77	0.78	0.76	0.59	0.71	0.68	0.29	0.75	0.73	0.48	0.83	0.80	0.73	0.57	0.54	0.12
SBERT*	0.40	0.38	0.03	0.86	0.85	0.73	0.88	0.87	0.68	0.82	0.80	0.36	0.86	0.85	0.61	0.86	0.83	0.53	0.64	0.61	0.09
CP-Tuning*	0.50	0.48	0.06	0.87	0.86	0.73	0.85	0.84	0.66	0.79	0.76	0.27	0.84	0.83	0.59	0.87	0.84	0.58	0.69	0.67	0.30
PromptSED*	0.51	0.50	0.12	0.89	0.88	0.79	0.87	0.86	0.71	0.83	0.81	0.30	0.86	0.85	0.60	0.88	0.86	0.66	0.70	0.68	0.31
HISEvent	0.38	0.37	0.09	0.90	0.89	0.88	0.90	0.89	0.79	0.77	0.76	0.52	0.83	0.82	0.63	0.89	0.88	0.84	0.64	0.63	**0.36**
RagSED	**0.55**	**0.53**	**0.27**	**0.95**	**0.95**	**0.93**	**0.95**	**0.95**	**0.94**	**0.90**	**0.89**	**0.68**	**0.93**	**0.93**	**0.90**	**0.97**	**0.96**	**0.97**	**0.75**	**0.73**	0.26
Promotion	↑.04	↑.03	↑.15	↑.05	↑.06	↑.05	↑.05	↑.06	↑.15	↑.07	↑.08	↑.16	↑.07	↑.08	↑.27	↑.08	↑.08	↑.13	↑.05	↑.05	↓.10

Blocks	M_8			M_9			M_{10}			M_{11}			M_{12}			M_{13}			M_{14}		
Metric	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI
KPGNN*	0.80	0.76	0.60	0.74	0.71	0.46	0.80	0.78	0.70	0.74	0.71	0.49	0.68	0.66	0.48	0.69	0.67	0.29	0.69	0.65	0.42
QSGNN*	0.79	0.75	0.59	0.77	0.75	0.47	0.82	0.80	0.71	0.75	0.72	0.49	0.70	0.68	0.49	0.68	0.66	0.29	0.68	0.66	0.41
SBERT*	0.88	0.86	0.65	0.85	0.83	0.47	0.87	0.85	0.62	0.84	0.82	0.49	0.86	0.85	0.63	0.73	0.70	0.24	0.79	0.77	0.40
CP-Tuning*	0.85	0.82	0.59	0.82	0.80	0.48	0.85	0.83	0.69	0.80	0.78	0.48	0.80	0.79	0.46	0.72	0.70	0.36	0.77	0.75	0.51
PromptSED*	0.87	0.85	0.59	0.86	0.83	0.56	0.86	0.85	0.71	0.84	0.82	0.51	0.86	0.85	0.52	0.74	0.71	0.54	0.80	0.78	0.53
HISEvent	0.82	0.81	0.68	0.89	0.88	**0.65**	0.91	0.90	0.87	0.85	0.84	0.66	0.87	0.87	0.82	0.75	0.74	0.39	0.83	0.82	0.71
RagSED	**0.96**	**0.95**	**0.92**	**0.91**	**0.90**	**0.65**	**0.96**	**0.96**	**0.97**	**0.95**	**0.94**	**0.96**	**0.91**	**0.91**	**0.83**	**0.88**	**0.87**	**0.77**	**0.93**	**0.92**	**0.92**
Promotion	↑.08	↑.09	↑.24	↑.02	↑.02	-	↑.05	↑.06	↑.10	↑.10	↑.10	↑.30	↑.04	↑.04	↑.01	↑.13	↑.13	↑.23	↑.10	↑.10	↑.21

Blocks	M_{15}			M_{16}			M_{17}			M_{18}			M_{19}			M_{20}			M_{21}		
Metric	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI	NMI	AMI	ARI
KPGNN*	0.58	0.54	0.17	0.79	0.77	0.66	0.70	0.68	0.43	0.68	0.66	0.47	0.73	0.71	0.51	0.72	0.68	0.51	0.60	0.57	0.20
QSGNN*	0.59	0.55	0.17	0.78	0.76	0.65	0.71	0.69	0.44	0.70	0.68	0.48	0.73	0.70	0.50	0.73	0.69	0.51	0.61	0.58	0.21
SBERT*	0.70	0.67	0.17	0.81	0.78	0.50	0.78	0.77	0.35	0.82	0.81	0.52	0.84	0.83	0.54	0.83	0.80	0.52	0.72	0.70	0.24
CP-Tuning*	0.64	0.61	0.31	0.80	0.77	0.74	0.76	0.75	0.58	0.76	0.75	0.44	0.82	0.81	0.43	0.83	0.79	0.55	0.71	0.69	0.40
PromptSED*	0.67	0.64	0.34	0.82	0.81	0.75	0.80	0.79	0.61	0.82	0.80	0.49	0.84	0.83	0.50	0.86	0.83	0.57	0.74	0.72	0.41
HISEvent	0.69	0.67	0.27	0.87	0.86	0.83	0.77	0.76	0.56	0.74	0.73	0.64	0.85	0.84	0.60	0.82	0.80	**0.67**	0.73	0.73	**0.46**
RagSED	**0.92**	**0.91**	**0.97**	**0.92**	**0.92**	**0.86**	**0.93**	**0.93**	**0.95**	**0.87**	**0.86**	**0.69**	**0.90**	**0.89**	**0.76**	**0.87**	**0.84**	**0.67**	**0.78**	**0.76**	0.38
Promotion	↑.22	↑.24	↑.63	↑.05	↑.06	↑.03	↑.13	↑.14	↑.34	↑.05	↑.05	↑.05	↑.05	↑.05	↑.16	↑.01	↑.01	-	↑.04	↑.03	↓.08

Table 2: Ablation results for SED. The best results are bolded, the second-best are underlined.

Blocks	M_3(Event2012)			M_{10}(Event2012)		
Metric	NMI	AMI	ARI	NMI	AMI	ARI
RagSED w/o KMS	.84	.81	.61	.93	.92	.88
RagSED w/o $\mathcal{KB}$	.86	.85	.57	.91	.89	.72
RagSED w/o KBM	.95	**.95**	.94	.96	.96	**.97**
RagSED (deepseek-r1:32b)	.95	**.95**	.94	.96	.96	**.97**
RagSED + deepseek-r1:70b	**.96**	**.95**	**.95**	.96	.96	**.97**
RagSED + GPT-4o-mini	**.96**	**.95**	**.95**	**.97**	**.97**	**.97**
RagSED + GPT-4o	**.96**	**.95**	**.95**	**.97**	.96	**.97**

guidance, enhancing the LLM's capability in SED; and the KBM module maintains the freshness of the knowledge base, which becomes particularly important for larger message blocks. Finally, we ablate RagSED with different base LLMs. While larger LLMs naturally achieve higher detection accuracy, RagSED with a 32B LLM already delivers competitive performance while significantly reducing computational resource requirements.

4 Conclusion

In this paper, we propose RagSED, a novel unsupervised social event detection model. First, RagSED introduces a key message sampling module that improves the quality of LLM queries while substantially reducing computational overhead. Second, RagSED designs a new RAG-based event detection paradigm, where the knowledge base provides global semantic guidance for detection. Extensive experiments on a benchmark dataset demonstrate that RagSED outperforms all baselines. In future work, we plan to extend this framework to multimodal scenes, further enhancing its utility.

References

[1] Yuwei Cao, Hao Peng, Jia Wu, Yingtong Dou, Jianxin Li, and Philip S Yu. 2021. Knowledge-preserving incremental social event detection via heterogeneous gnns. In *Proceedings of the web conference 2021*. 3383–3395.

[2] Yuwei Cao, Hao Peng, Zhengtao Yu, and Philip S Yu. 2024. Hierarchical and incremental structural entropy minimization for unsupervised social event detection. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 38. 8255–8264.

[3] Pablo A Estévez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. 2009. Normalized mutual information feature selection. *IEEE Transactions on neural networks* 20, 2 (2009), 189–201.

[4] Dan Knights, Michael Mozer, and Nicolas Nicolov. 2009. Detecting topic drift with compound topic models. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 3. 242–245.

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv:2005.11401 [cs.CL] https://arxiv.org/abs/2005.11401

[6] Andrew J McMinn, Yashar Moshfeghi, and Joemon M Jose. 2013. Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 409–418.

[7] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3982–3992.

[8] Jiaqian Ren, Lei Jiang, Hao Peng, Yuwei Cao, Jia Wu, Philip S Yu, and Lifang He. 2022. From known to unknown: Quality-aware self-improving graph neural network for open set social event detection. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 1696–1705.

[9] Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2009. Information theoretic measures for clusterings comparison: is a correction for chance necessary?. In *Proceedings of the 26th annual international conference on machine learning*. 1073–1080.

[10] Ziyun Xu, Chengyu Wang, Minghui Qiu, Fuli Luo, Runxin Xu, Songfang Huang, and Jun Huang. 2023. Making pre-trained language models end-to-end few-shot learners with contrastive prompt tuning. In *Proceedings of the sixteenth ACM international conference on web search and data mining*. 438–446.

[11] Xiaoyan Yu, Jiaqian Ren, Lei Jiang, Hao Peng, Zhifeng Hao, Li Sun, Kun Peng, Liehuang Zhu, and Philip S Yu. 2025. PromptSED: An evolving topic-enhanced prompting framework for incremental social event detection. *Neural Networks* (2025), 107772.

[12] Xiaoyan Yu, Yifan Wei, Shuaishuai Zhou, Zhiwei Yang, Li Sun, Hao Peng, Liehuang Zhu, and Philip S Yu. 2025. Towards effective, efficient and unsupervised social event detection in the hyperbolic space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 13106–13114.