From Known to Unknown: Quality-aware Self-improving Graph Neural Network for Open Set Social Event Detection

Jiaqian Ren[1,2], Lei Jiang[2], Hao Peng[3], Yuwei Cao[4], Jia Wu[5], Philip S. Yu[4], Lifang He[6]
[1] Institute of Information Engineering, Chinese Academy of Sciences;
[2]School of Cyber Security, University of Chinese Academy of Sciences;
[3]Beihang University; [4]University of Illinois Chicago;[5]Macquarie University; [6]Lehigh University.
* Corresponding authors (Jianglei@iie.ac.cn, penghao@buaa.edu.cn)

ABSTRACT

State-of-the-art Graph Neural Networks (GNNs) have achieved tremendous success in social event detection tasks when restricted to a closed set of events. However, considering the large amount of data needed for training a neural network and the limited ability of a neural network in handling previously unknown data, it remains a challenge for existing GNN-based methods to operate in an open set setting. To address this problem, we design a Quality-aware Self-improving Graph Neural Network (QSGNN) which extends the knowledge from known to unknown by leveraging the best of known samples and reliable knowledge transfer. Specifically, to fully exploit the labeled data, we propose a novel supervised pairwise loss with an additional orthogonal inter-class relation constraint to train the backbone GNN encoder. The learnt, already-known events further serve as strong reference bases for the unknown ones, which greatly prompts knowledge acquisition and transfer. When the model is generalized to unknown data, to ensure the effectiveness and reliability, we further leverage the reference similarity distribution vectors for pseudo pairwise label generation, selection and quality assessment. Following the diversity principle of active learning, our method selects diverse pair samples with the generated pseudo labels to fine-tune the GNN encoder. Besides, we propose a novel quality-guided optimization in which the contributions of pseudo labels are weighted based on consistency. We thoroughly evaluate our model on two large real-world social event datasets. Experiments demonstrate that our model achieves state-of-the-art results and extends well to unknown events.

CCS CONCEPTS

• **Computing methodologies → Artificial intelligence**; • **Information systems → Data mining**.

KEYWORDS

Social event detection, graph neural network, contrastive learning, active learning

ACM Reference Format:
Jiaqian Ren[1,2], Lei Jiang[2], Hao Peng[3], Yuwei Cao[4], Jia Wu[5], Philip S. Yu[4], Lifang He[6]. 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 and Knowledge Management (CIKM'22).* ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The goal of social event detection is to discover meaningful events from the social media data by grouping messages reported on the same event together [3, 21]. Due to its broad range of potential applications such as business marketing, disaster risk management, public opinion analysis, etc., social event detection has experienced explosive growth in recent years [10].

Previous studies often utilize text contents [2, 27, 38, 45, 49, 50] or auxiliary attributes extracted from texts [11, 47, 48] to make social event detection. However, in recent years, there is a trend towards GNN-based methods [6, 7, 30–32] which have the ability to combine contents and attributes together. By fully capturing the rich semantics and structural information contained in the social data, GNN-based approaches achieve remarkable performance when restricted to a closed set of events. Taking a deep dive into these neural network models, their success heavily relies on the huge amount of data with human annotations under the closed-world assumption. However, the data of social networks is being updated all the time, which means there are lots of newly emerging events. On the one hand, consistently identifying and annotating all emerging categories for model training are cost-inhibitive. On the other hand, given the presence of significant distribution gaps between the already-known training events and the emerging unknown events, directly applying the models trained on known data to unknown data in the real world typically exhibits clear performance degradation. When the conventional closed set assumption no longer holds, how to keep the strength of GNN methods as well as generalize them to open set application is a challenging problem.

We argue that the aforementioned problem can be solved by human-like learning processes. A human can easily identify samples of new events after they are trained to distinguish events with some already-known samples. Likewise, for the training of models, it is natural to leverage the already-known events (the annotated data) to explore the new, unlabeled events, rather than in a completely unsupervised way. Generally, the success of human beings in recognizing new events is due to two kinds of abilities: 1) the ability to find the pattern of an event with already-known samples, and 2) the ability to transfer knowledge from known to unknown.

CIKM'22, October 17-22, 2022, Hybrid Conference, Hosted in Atlanta, Georgia, USA

ACM ISBN 978-1-4503-XXXX-X/18/06.
https://doi.org/XXXXXXX.XXXXXXX

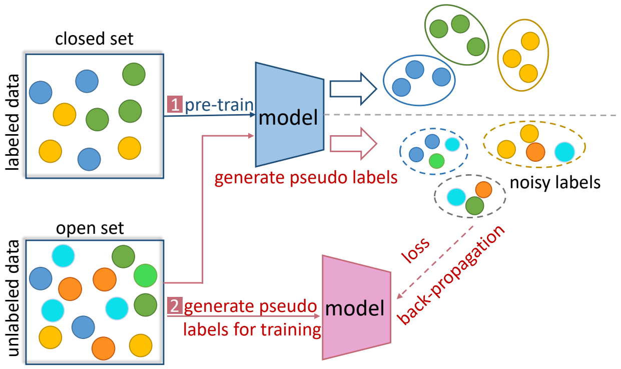


Figure 1: The two-step learning strategy for knowledge transferring from known to unknown.

Analogously, the key challenges in open set social event detection are: 1) how to fully exploit the labeled data to learn discriminative features, and 2) how to achieve effective and reliable knowledge transfer to promote the discovery of new events. However, existing GNN-based event detection methods still have room for improvement on the first issue, and totally ignore the second one. Consider the first challenge, authors in [7, 30] train GNN networks to learn the discriminative features of events based on the cross-entropy loss. While great results have been achieved, they are unsuitable for the open world. Authors in [6] and [32] replace the cross-entropy loss with a triplet loss to train the model. However, they still cannot fully exploit data. This is because the complicated sampling strategy in the triplet loss brings a weaker generalization capability when being extended to the unknown set, and the relative distances between positive and negative pairs learnt from the triplet loss are not always distinguishable (the intra-class distances are sometimes larger than the inter-class distances). To solve these problems, we propose a stricter pairwise loss in this paper.

As to the second challenge, i.e., knowledge transferring from known to unknown, those GNN-based methods [6, 32] which consider the incremental, novel events setting in the real world, need labeled data for continuous training. However, constantly labeling new data is time-consuming and labor-intensive. To achieve knowledge transfer in an unsupervised way, it is worth noting that a series of methods [17–20, 51, 52] are proposed for the task of novel class discovery. As illustrated in Fig. 1, these methods commonly follow a two-step learning strategy: 1) pre-training the model with labeled data to obtain basic discriminative ability; 2) adopting clustering or pairwise determination algorithms to generate pseudo labels for the training of the new classes. However, using all pseudo labels for training is unnecessary and time-consuming. Besides, these methods fail to consider the noise in the obtained pseudo-labels and, therefore, are unreliable. How to select a small number of high-quality samples to achieve effective and reliable knowledge transfer becomes an important problem. In a word, the task of open set social event detection is quite hard and has not been well-solved yet.

To tackle the above challenges, in this work, we design a Quality-aware Self-improving Graph Neural Network (QSGNN) which extends the knowledge from known to unknown by leveraging the best of the known samples and reliable knowledge transfer. Specifically, to fully explore the discriminative knowledge from the already-known data, we propose a novel pairwise learning method which has a stricter demand on the distances between intra-class samples and inter-class ones to train our backbone GNN encoder. In addition to differentiating the known events by the distances in the latent space, we also add an orthogonal inter-class constraint to force the known events to scatter in different directions. In this way, those known events are fully explored and become strong reference bases for unknown events. To ensure the effectiveness and reliability of the knowledge transferring process, we propose to utilize the Reference Similarity Distribution (RSD) vector, which is obtained by computing the similarity distribution over a set of known reference events, for pseudo pairwise label generation, selection and quality assessment. Particularly, we follow the principle of diversity in active learning [36] to make unbalanced sample selection, and assess the quality of the pseudo labels by their consistencies. To further resist the noise of the pseudo labels, we adopt a quality-guided optimization in which the contributions of the the pseudo labels are weighted. We continuously fine-tune the backbone GNN encoder to adapt to the incremental, unseen events in the open world. Note that our model provides a principled way to obtain pairwise pseudo labels with high quality for unlabelled data, thus enabling effective and reliable knowledge transfer.

Experimental results on two large and publicly available Twitter datasets show that our method achieves state-of-the-art performance in closed set event detection and maintains high performance in the open set setting. The source code and data are available at GitHub[1]. Our main contributions are summarized as follows: 1) We propose a Quality-aware Self-improving Graph Neural Network (QSGNN) which extends knowledge from known to unknown by leveraging the best of known samples and reliable knowledge transfer. It successfully solves the open set social event detection problem. 2) We propose a novel pairwise learning method with an orthogonal inter-class constraint. Our method demands stricter distance distinctions as well as direction distinctions in the latent space, thus fully exploits the knowledge contained in the labeled data. 3) We ensure effective and reliable knowledge transfer from known to unknown by a selection strategy based on diversity as well as a quality assessment strategy based on consistency. The effectiveness comes from the unbalanced sample selection, and the reliability is guaranteed by the quality-guided optimization process.

2 RELATED WORK

2.1 Social Event Detection

Event detection in social networks has received considerable attention in recent years [15]. Based on existing techniques, social event detection methods can be divided into incremental clustering ones [1, 29], community detection ones [10, 22], and topic modeling ones [47–50]. Though the incremental clustering methods can be easily adapted to open set social event detection tasks, they fail to fully explore the knowledge contained in the social streams as they ignore the rich semantics and structural information. This

[1] https://github.com/RingBDStack/open-set-social-event-detection

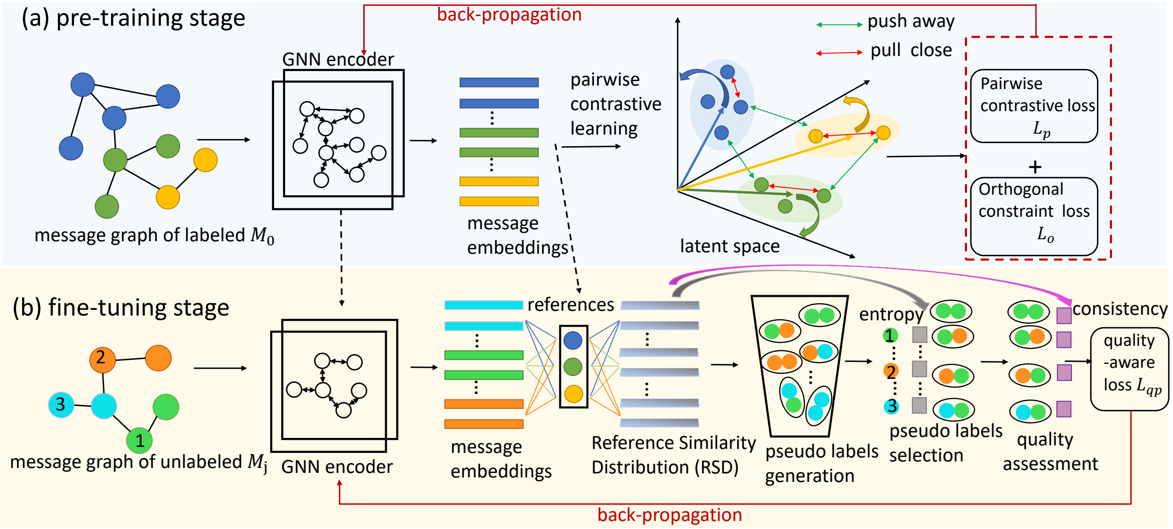


Figure 2: The architecture of QSGNN. (a) shows the pre-training stage, in which we fully explore the discriminative knowledge from the labeled data by utilizing both distance and direction information. (b) demonstrates the fine-tuning stage, in which we utilize the Reference Similarity Distribution vector for pseudo pairwise label generation, selection and quality assessment.

problem also exists in the community detection methods and the topic modelling methods. Based on the information they exploit, social event detection methods can be categorized into three types: content-based [2, 27, 38, 45, 49, 50], attribute-based [11, 47, 48], and combined methods [6, 7, 24, 30–32, 44]. Among the combined methods, the GNN-based ones [6, 7, 30–32, 34, 35] perform greatly due to their powerful expressive ability in building social graphs to effectively combine contents and attributes. However, most GNN-based studies hold the closed-set assumption in which the training set and test set share the same events thus cannot be directly applied in the open world. Few exceptions [6, 31, 32] do consider the incremental events setting but still need annotated data for continuous training, which is costly. Considering the limited ability of neural networks in handling previously unknown data, adapting GNN-based methods to the open world setting remains challenging.

2.2 Active Learning

Active learning, also known as query learning, assumes that different samples in the dataset have different values for the model training, and tries to select a small quantity of data to achieve high performance gains [36]. Based on the query principle they employ, active learning methods can be split into three categories: uncertainty-based methods [4, 23, 33, 42], diversity-based methods [5, 13, 16, 28, 46], and expected model change-based ones [12, 14, 37, 40]. In this work we follow the principle of diversity and propose an unbalanced sampling strategy.

2.3 Novel Class Discovery

The task of novel class discovery is proposed recently aiming at recognizing novel classes in unlabeled data [18], which is similar to our open set event detection task. This task differs from the conventional unsupervised learning as they leverage some already-known data. Existing methods [17–20, 51, 52] usually follow a two-step strategy: 1) using the labeled data for model initialization, 2) performing unsupervised clustering or pairwise determination on unlabeled data to fine-tune the model. For example, to recognize open set images, authors in [19] propose a constrained clustering network. They first measure the pairwise similarities between images by training a classification model on labeled data, then adopt a clustering model on unlabeled data with those pairwise predictions. Later, authors in [17] directly utilize rank statistics to estimate the pairwise similarity of images. Though these approaches which use pseudo labels to achieve model adaption to the unlabelled data have achieved promising results, they fail to consider the noise contained in the obtained pseudo-labels. Therefore, the training process is unreliable. What's more, they have not given a sample selection strategy. Considering the large amount of unlabeled data, how to prompt performance with a small number of samples becomes an important problem.

3 METHODOLOGY

In this section, we begin by presenting the problem definition of the open set social event detection task in Sec. 3.1, and then introduce the details of our model. Fig. 2 shows an overview of our proposed QSGNN, which includes the supervised pre-training stage (Sec. 3.3) and the self-improving fine-tuning stage (Sec. 3.4).

3.1 Problem Definition

Here we give the definition of the open set social event detection task in this work. The goal of our work is to identify newly emerging events in unlabelled incremental social streams with the support of knowledge learnt from the existing known events.

Formally, the open set event data comes as incremental social stream [6]. The social stream, denoted as $S = M_0, M_1, ..., M_{i-1}, M_i$, is actually a temporal sequence of blocks of social messages, where each message block M_j, $j \in \{0, 1, ..., i\}$ contains all the messages that arrive during the split time period. Different from [6], in this work we impose the constraint that only the messages in the initial block (i.e., M_0) are provided with labels. The later blocks remain unlabeled during the whole training process. The objective is two-fold: 1) In the pre-training stage, from the labeled message block M_0, we learn an initial detection encoder, $f(M_0, \theta_0)$, that extracts discriminative features and detects events. θ_0 denotes the parameter of $f(M_0, \theta_0)$ and is trained from M_0. 2) To detect events from the unlabeled message blocks, we extend knowledge from known to unknown by consistently updating the detection encoder. That is, we learn a sequence of detection encoders $f(M_j, \theta_0, \theta_j), j \in \{1, ..., i\}$. Each of them is fine-tuned on the corresponding unlabeled message block to achieve knowledge transferring.

3.2 Overall Framework

Our work contains two stages: 1) the model pre-training stage with supervised pairwise contrastive learning (Eq. (3)), and 2) the model fine-tuning stage with quality-aware self-improving learning (Eq. (8)). As mentioned in Sec. 3.1, we split the whole dataset into an incremental social stream. In the pre-training stage, an initial message graph is constructed. We assume those messages are all labeled and utilize them to train our backbone GNN encoder. In the fine-tuning stage, messages are all assumed to be unlabeled. For each message block, we construct a new graph. We directly input each coming block to the already trained backbone model to get initial representations for pseudo labels generation, selection, and quality assessment. Note that for each unlabeled block, we consistently update the model with the generated pseudo labels and re-generate new pseudo labels for three times. In this way, the model adapts to the incoming data. Besides, utilizing only the current message block for fine-tuning, our procedure maintains a light training scheme.

3.3 Supervised Pre-training of Backbone GNN Encoder

3.3.1 *Construction of message graph.*

Here we give a brief introduction to the message graph construction process. The message graph, which only contains message nodes, is used to express the complicated internal relations of messages in the social stream. To fully explore all kinds of important information contained in the social stream, we build edges between messages based on three types of elements including users, hashtags, and entities. Messages share any of these three elements are linked together. As for the initial message features, we follow the way in [6] which combines the language semantics with temporal information together. Specifically, the semantic feature of a message is obtained by averaging the pre-trained embeddings [41] of its words. The 2-d temporal feature is obtained by converting the timestamp to the OLE date. The initial message feature is the concatenation of these two parts.

3.3.2 *Backbone GNN encoder.*

After constructing the message graph, we apply a GNN encoder on it to learn comprehensive message representations. To fully incorporate the rich semantics and relations, the GNN encoder learns node representations by iteratively combining information from their one-hop neighbours. Formally, for message m_i, whose representation in the l-th layer is denoted as $\boldsymbol{h}^l_{m_i}$, its updated representation in the $(l+1)$-th layer becomes:

$$\boldsymbol{h}^{(l+1)}_{m_i} \leftarrow \overset{\text{heads}}{\|} \left(\boldsymbol{h}^{(l)}_{m_i} \oplus \underset{\forall m_j \in \mathcal{N}(m_i)}{\text{Aggregator}} \left(\text{Extractor} \left(\boldsymbol{h}^{(l)}_{m_j} \right) \right) \right), \quad (1)$$

where $\mathcal{N}(m_i)$ denotes the one-hop neighbours of message m_i. $\oplus$ stands for an aggregation, and $\overset{\text{heads}}{\|}$ represents concatenation of multiple heads. Aggregator and Extractor strategies differ in different GNN models. In this paper we adopt the strategies in GAT [43].

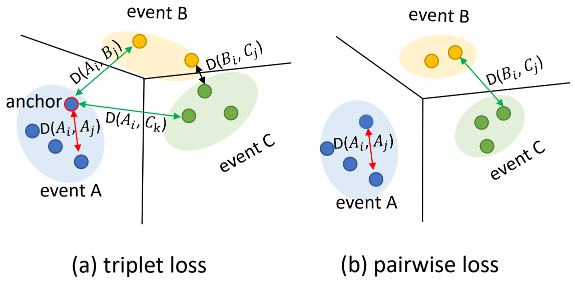


Figure 3: (a) and (b) illustrate the representation distributions learnt by the triplet loss vs. the pairwise loss (ours). Our pairwise loss guarantees that the intra-class distances are always smaller than the inter-class ones, e.g., $D(A_i, A_j) < D(B_i, C_j)$.

3.3.3 *Supervised pairwise contrastive learning.*

As new events continuously arrive in the open world, the total number of events is hard to know in advance. Thus, cross-entropy loss functions that are widely used in the closed set are not suitable anymore. To cope with the discrepancies between the closed and the open set settings, we need to design a training method which can perform well under supervision as well as be easily generalized to unknown samples. Accordingly, we propose a novel pairwise learning method, which is inspired by the triplet loss in [39], but different in the removal of anchors. The essence of our pairwise learning is the idea that the distance between an intra-class pair should be smaller than any of the inter-class pairs, no matter whether they share a common node or not.

In [39], to compute the triplet loss, one must construct a set of triplets $\{T\}$ first. Each triplet is composed of an anchor, a positive sample to the anchor (i.e., a sample within the same class), and a negative sample to the anchor (i.e., a sample from a different class). Suppose the anchor message is m_i. m_i+ and m_i- denote the positive and the negative samples, respectively. The triplet loss is as follows:

$$\mathcal{L}_t = \sum_{(m_i, m_i+, m_i-) \in \{T\}} \max \left\{ \mathcal{D}\left(\boldsymbol{h}_{m_i}, \boldsymbol{h}_{m_i+}\right) - \mathcal{D}\left(\boldsymbol{h}_{m_i}, \boldsymbol{h}_{m_i-}\right) + a, 0 \right\}, \quad (2)$$

where a is a hyper-parameter which represents the margin distance. $D(\cdot, \cdot)$ computes the Euclidean distance.

Carefully analysing the triplet loss computation process mentioned above, we find two deficiencies: 1) Its sampling strategy is complicated, which hampers its generalization from the known set to the unknown set. When the labels are provided, one can easily obtain triplets by constructing all possible positive pairs in a specific class and adding the same number of random negative samples from the other classes. However, for unknown data, the triplets are much harder to construct. Specifically, all node pairs need to be classified before we can combine the positive ones with other samples to form triplets. 2) The triplet loss which trains the model based on the relative distances between positive and negative pairs w.r.t. the same anchor is a relaxed constraint. As shown in Fig. 3(a), after training, the intra-class distance in event A may be larger than the inter-class distance between event B and event C. E.g., $D(B_i, C_j)$ is smaller than $D(A_i, A_j)$, which creates difficulties for the correct clustering of events.

To simplify the sampling process and make the intra-class distributions more distinguishable from the inter-class ones, our pairwise loss introduces a stricter constraint which pushes away negative pairs from positive pairs despite the anchors. The loss is as follows:

$$\mathcal{L}_p = \sum_{\substack{(m_i,m_i+)\in\{Pos\}\\(m_j,m_{j-})\in\{Neg\}}} \max\{\mathcal{D}(\boldsymbol{h}_{m_i}, \boldsymbol{h}_{m_i+}) - \mathcal{D}(\boldsymbol{h}_{m_j}, \boldsymbol{h}_{m_j-}) + a, 0\}, \tag{3}$$

where $\{Pos\}$ denotes the positive pairs and $\{Neg\}$ denotes the negative ones. Obviously, with the help of this loss, the minimum inter-class distance in the whole batch is required to be larger than the maximum intra-class distance, as shown in Fig. 3(b). This helps differentiate the intra-class representations from the inter-class ones thus improves the events clustering results.

3.3.4 Orthogonal inter-class relation constraint. The pairwise loss distinguishes the events by solely controlling the distances between messages in the latent space. There is no utilization of the direction information. Therefore, the learnt representations of messages belonging to different events may concentrate in one direction and results in a certain dimension "waste". As shown in Fig. 2(a), to fully leverage the latent space and maximally differentiate the known events, we force the learnt features of events in different classes to be scattered in different directions by adding an orthogonal constraint. Specifically, we build a target pairwise similarity matrix based on the ground-truth events labels and demand the cosine similarity of the learnt message representations be close to it. Suppose we have N_b nodes in a batch, the target pairwise similarity matrix is $\mathbf{P} \in \mathbb{R}^{N_b \times N_b}$. $\mathbf{P}_{ij}$ is 1 if message m_i and message m_j belong to the same event; otherwise, the value is 0. The cosine similarities of the learnt message representations are computed as $\overline{\boldsymbol{H}} \cdot \overline{\boldsymbol{H}}^T$, where $\overline{\boldsymbol{H}} \in \mathbb{R}^{N_b \times d}$ denotes the normalized message representations. The additional orthogonal loss is:

$$\mathcal{L}_o = \mathrm{Sum}((\boldsymbol{P} - \overline{\boldsymbol{H}} \times \overline{\boldsymbol{H}}^T)^2), \tag{4}$$

where $\mathrm{Sum}(\cdot)$ represents the sum of the elements in the matrix.

Obviously, with the orthogonal inter-class relation constraint, directions are also utilized to train the message representations, which further differentiates the events in the latent space. Besides, with the orthogonal constraint, the known events become strong reference bases for those unknown samples thus greatly prompts knowledge transfer.

3.4 Self-improving Fine-tuning of GNN Encoder

3.4.1 Pseudo Pairwise Labels Generation. Previous works [17, 19] often use the trained model to get the initial representations of unknown data and generate pseudo labels based on their similarities. However, for those newly emerging events, it is very challenging to obtain discriminative representations due to their absence during the training process. Therefore, labels learnt from capturing merely the local data similarities are often of low quality.

As illustrated in Fig. 2(b), to overcome this problem, we propose to utilize the Reference Similarity Distribution (RSD) vectors to generate pseudo labels. The idea is to learn soft multilabel vectors (real-valued label likelihood vectors) for those unknown samples by computing their similarities to a set of known reference events. Specifically, assume that $\overline{\boldsymbol{R}} = \{\overline{\boldsymbol{r}}_{e1}, \overline{\boldsymbol{r}}_{e2}, ...\overline{\boldsymbol{r}}_{eK}\} \in \mathbb{R}^{K \times d}$ denotes a matrix which stores the normalized representations of K known reference events, with each row denoting a reference event. For example, $\overline{\boldsymbol{r}}_{ek} \in \mathbb{R}^d$ is the normalized cluster center of the k-th event. Suppose the normalized representation of a message sample m_i from unknown data is denoted as $\overline{\boldsymbol{h}}_{m_i}$, its RSD vector $\boldsymbol{p}_{m_i} \in \mathbb{R}^K$ is calculated as:

$$\boldsymbol{p}_{m_i} = \mathrm{Softmax}(\overline{\boldsymbol{h}}_{\boldsymbol{m_i}} \cdot \overline{\boldsymbol{R}}^T). \tag{5}$$

The RSD vector, $\boldsymbol{p}_{m_i}$, captures the global similarities between m_i and those known events. It contains much more knowledge compared to $\overline{\boldsymbol{h}}_{\boldsymbol{m_i}}$, the originally learnt representation and thus has stronger discriminative power. Meanwhile, it is worth noting that those reference events are mutually discriminated from each other under the orthogonal constraint. This guarantees the effectiveness of them as strong reference bases.

We use the cosine similarity between the RSD vectors of two messages to measure their consistency and generate the pseudo label for the pair. Suppose there is a pair of messages (m_i, m_j), the consistency value $C(m_i, m_j)$ is calculated as:

$$C(m_i, m_j) = \frac{\boldsymbol{p}_{m_i} \cdot (\boldsymbol{p}_{m_j})^T}{||\boldsymbol{p}_{m_i}||_2 \cdot ||(\boldsymbol{p}_{m_j})^T||_2}, \tag{6}$$

where $|| \cdot ||_2$ denotes the ℓ_2 norm. If $C(m_i, m_j) > 0.5$, we set the pseudo label to 1 (positive); otherwise, we set it to 0 (negative).

3.4.2 Pseudo Pairwise Labels Selection. According to the pseudo pairwise label generation strategy described above, we get candidate pseudo labels of all possible pairs in a batch. Using all of them to fine-tune the model is both inefficient and unnecessary, because there are unavoidably numerous misclassified labels. How to select a small number of informative samples to achieve effective and efficient event knowledge transfer becomes an important problem. Inspired by the active learning [46] which uses diversity as an indicator of sampling, we actively select a small number of pairs which are most different from the known data. In this way, the generalization ability of the learned model can be significantly strengthened with a limited training cost.

In the unknown data, there are some samples which belong to the known events and some which are previously unseen. We have observed from experiments (Sec. 4.3.3) that the RSD vectors of samples in new events have higher information entropy compared to those within the known events. That's because a RSD vector, in its essence, is a soft multilabel probability vector. For those messages that belong to the known events, their RSD vectors tend to be one-hot vectors whose information entropy values are close to 0. However, for messages within the new events, their distributions may concentrate in the space between several events, and thus the entropy is relatively large. Based on this observation, we propose to exploit the entropy to estimate the diversity of a message sample. Here, diversity refers to the degree of difference from the known events used to train the backbone model. To ensure effective knowledge transfer, we focus on the samples that are different from the known events during the fine-tuning stage. Therefore, for each message, we use the entropy of its RSD vector to determine the number of its pairing messages. Formally, for message m_i, whose RSD vector is $\boldsymbol{p}_{m_i}$, the entropy is:

$$H(\boldsymbol{p}_{m_i}) = -\sum_{j=1}^{K} \boldsymbol{p}_{m_i j} \log_2 \left(\boldsymbol{p}_{m_i j}\right). \tag{7}$$

We employ an unbalanced sampling strategy and split the messages into two groups based on their entropy values. In experiments, for half of messages with larger entropy values, we sample 20 negative pairs and 20 positive pairs for model fine-tuning. For the other half, we sample 10 negative and 10 positive pairs.

3.4.3 *Pseudo Pairwise Labels Quality Assessment.*

The quality assessment of those selected pseudo labels is also important. Without proper assessment and selection, the noisy pseudo-labels will gradually undermine the model. We have observed from experiments (Sec. 4.3.4) that the consistency values (i.e. the cosine similarities between the RSD vectors) of pairs with wrong pseudo-labels are closer to the dividing threshold 0.5. Based on this observation, we simply exploit the consistency value to estimate the quality of the pseudo labels. For positive pairs whose values are larger than 0.5, the quality is positively correlated with consistency. On the contrary, for negative pairs whose values are smaller than 0.5, the quality is negatively correlated with consistency. For simplicity, we directly assign the consistency value, $C(\mathrm{Pos})$, of those pseudo positive pairs (consistency > 0.5) as the quality, and assign $1 - C(\mathrm{Neg})$ as the quality of those pseudo negative ones, where Pos and Neg stand for a positive or negative message pair.

3.4.4 *Quality-guided Optimization.*

To make the self-training (fine-tuning) process focus more on pair samples that the model is more confident on, we re-weight the importance of the pairs based on their quality. The quality-guided pairwise contrastive loss becomes:

$$\begin{aligned}\mathcal{L}_{qp} = \sum_{\substack{(m_i, m_i+)\in\{Pos\}\\(m_j, m_{j-})\in\{Neg\}}} & (C(m_i, m_i+) + 1 - C(m_j, m_j-))\cdot \\ & \max\{\mathcal{D}(\boldsymbol{h}_{m_i}, \boldsymbol{h}_{m_i+}) - \mathcal{D}(\boldsymbol{h}_{m_j}, \boldsymbol{h}_{m_j-}) + a, 0\}.\end{aligned} \tag{8}$$

During the fine-tuning process, we leverage the quality-guided pairwise contrastive loss computed from those selected pair samples to update the model.

After model fine-tuning, we utilize the updated model to get the representations of all the unknown samples and adopt a specific clustering algorithm to output a set of social events. As for the clustering method, we can select distance-based clustering algorithms such as K-means or density-based ones such as DBSCAN [9]. Note that DBSCAN does not require specifying the total number of classes and thus is more suitable for open set social event detection.

4 EXPERIMENTS

4.1 Experimental Setup

4.1.1 *Datasets.*

We evaluate QSGNN on two large publicly available social event datasets: Events2012 [26] and Events2018 [25]. We crawl the tweets via the Twitter API based on the provided IDs. After filtering out unavailable tweets, Events2012 contains 68,841 annotated tweets belonging to 503 event classes, and spreads over a period of 4 weeks. As for Events2018, it has 64,516 labeled tweets belonging to 257 event classes within 4 weeks.

4.1.2 *Baselines.*

We compare QSGNN to both non-GNN-based methods and GNN-based methods. For the former, the baselines are: (1)**TwitterLDA** [50] which is the first proposed topic model for Tweet data; (2) **Word2Vec** [41], which uses the average of the pre-trained Word2Vec embeddings of all words in the message as its representation; (3) **BERT** [8], which uses the 768-d sentence embeddings of BERT as the message representations; (4) **EventX** [22], which detects events based on community detection. For GNN-based methods, we select (5) **PP-GCN** [30], an offline fine-grained social event detection method based on GCN. (6) **KPGNN** [6] which leverages triplet loss to train GAT and gets message representations.

4.1.3 *Implementation Details.*

Our QSGNN is built on the PyTorch framework and on a machine equipped with seven NVIDIA GeForce RTX 3090 GPUs. As for the specific GNN encoder adopted in this work, we select a 2-layer GAT network. We set the total number of heads to 4, the hidden and output embedding dimensions to 32, the learning rate to 0.001, optimizer to Adam. In the pre-training stage, we set the training epochs to 15 with a patience of 5 for early stopping. In the fine-tuning stage, we set the training epochs to 3. Besides, we set the batch size to 2000 and the distance margin a to 10. We repeat all experiments for 5 times and report the mean and standard deviation of the results. Some baselines (e.g., TwitterLDA, Bert) require the number of total event classes to be pre-defined. Thus, for a fair comparison, we apply the K-means clustering method after obtaining the message representations of all the models and set the total number of classes to the number of ground-truth classes. When applied in the real world, the K-means method can be replaced by DBSCAN, which does not require a pre-defined class number.

4.1.4 *Evaluation Metrics.*

The performances of models are evaluated by two widely used clustering metrics: normalized mutual information (NMI) and adjusted mutual information (AMI). By measuring the amount of information from the distribution of the predictions, NMI has been broadly adopted in event detection method evaluations. Meanwhile, considering NMI is not adjusted for chance, we also select AMI.

4.2 Evaluation on the Closed Set

Recall that our model contains two stages: the supervised pre-training stage and the self-improving fine-tuning stage. In the pre-training stage, we assume the event labels are all available and utilize those known data in the initial block to train the backbone GNN encoder. While in the fine-tuning stage, messages from the incoming blocks are assumed to be unknown. To validate the effectiveness of our novel supervised pairwise contrastive learning method with

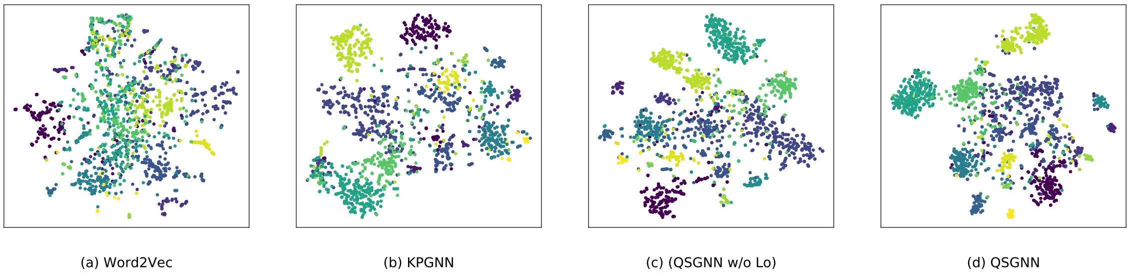

(a) Word2Vec
(b) KPGNN
(c) (QSGNN w/o Lo)
(d) QSGNN

Figure 4: Visualization on Events2018. Each node denotes a message and each color denotes an event class.

Table 1: Evaluation on the Closed Set.

Methods	M_0 in Events2012		M_0 in Events2018	
	NMI	AMI	NMI	AMI
TwitterLDA [50]	.26±.00	.17±.00	.22±.00	.16±.00
Word2Vec [41]	.47±.00	.21±.00	.24±.00	.20±.00
BERT [8]	.63±.01	.44±.00	.42±.00	.34±.00
EventX [22]	.68±.00	.29±.00	.57±.00	.56±.00
PP-GCN [30]	.70±.02	.56±.01	.60±.01	.49±.02
KPGNN [6]	.76±.02	.64±.02	.66±.03	.60±.02
QSGNN w/o $\mathcal{L}_o$	.77±.00	.65±.00	.68±.02	.61±.01
QSGNN	**.79±.01**	**.68±.01**	**.71±.02**	**.64±.02**
promotion	↑ 3%	↑ 4%	↑ 5%	↑ 4%

the orthogonal constraint, we compare the performances in the closed set situation in which the training set, validation set and the test set share the same events.

Specifically, for both Events2012 and Events2018 datasets, we use the data of the first week to form the initial message block M_0. We randomly sample 20% of the initial block for testing, 10% for validation, and use the rest 70% for training.

4.2.1 Comparison with the state-of-the-arts. As shown in Table 1, QSGNN yields the best results. That's because it fully utilizes space distance as well as direction information to distinguish different events. Compared to KPGNN, which only uses distance information to learn event representations, QSGNN achieves 3% and 5% performance gains in NMI on Events2012 and Events2018, respectively. It is worth noting that, even without the direction information, i.e., the orthogonal inter-class constraint, (QSGNN w/o $\mathcal{L}_o$) still works better than KPGNN. That is due to the more strict distance constraint. The triplet loss adopted in KPGNN only requires the intra-class distance to be smaller than the inter-class distance of the same anchor. However, the pairwise loss proposed in QSGNN demands the inter-class distance to be smaller than the minimum inter-class distance. Therefore, the intra-class representations learnt by (QSGNN w/o $\mathcal{L}_o$) are more distinguishable from the inter-class ones. Besides, we also notice that GNN-based methods (i.e., PP-GCN, KPGNN and QSGNN) perform much better than general message representation learning methods (i.e., Word2Vec and BERT) and word distribution methods like TwitterLDA. For example, QSGNN gets a large improvement (53%) in NMI compared to TwitterLDA in Events2012. We owe this contribution to the effectiveness of GNN-based methods in exploring the graph structure contained in the social network.

4.2.2 Visualization. For a more intuitive comparison and to further show the effectiveness of our proposed QSGNN, we conduct visualization on Events2018 by plotting the representations of the test set using t-SNE. The results are illustrated in Fig. 4. Obviously, GNN-based methods which capture both semantics and internal structure information are capable to learn more distinguishable representations compared to Word2Vec. Meanwhile, the observation that intra-class representations learnt by (QSGNN w/o $\mathcal{L}_o$) gather more closely compared to KPGNN verifies the effectiveness of our novel pairwise contrastive learning loss. Furthermore, the more concentrated intra-class distribution in QSGNN compared to (QSGNN w/o $\mathcal{L}_o$) demonstrates the effects of adding orthogonal constraint during training.

4.3 Evaluation on the Open Set

In the self-improving fine-tuning stage, we assume the data is unknown. To achieve knowledge transferring and make the model adapt to the incoming new data, our model generates pseudo labels to update the initial model. However, for most baselines such as PP-GCN and KPGNN, it is necessary to continuously provide event data with labels to train the model for the new message blocks. To compare with them, for those baselines which need supervision, we still follow the operation in the closed set - sampling 70% for training, 10% for validation and 20% for testing. Note that our model is superior to those methods since it does not require external annotation which is labour-costly.

4.3.1 Comparison with the state-of-the-arts. We demonstrate the results in Table 2 and Table 3. Generally, QSGNN outperforms the strongest baseline, KPGNN, in most message blocks (with 1%-4% performance gains). Note that the proposed QSGNN, unlike KPGNN and the other baselines, does not require ground-truth labels for continuous model training or updates. It is impressive that QSGNN, which gets fine-tuned only by the generated pseudo pairwise labels, performs even better than those supervised baselines. This varifies the superiority and effectiveness of our model in extending knowledge from known to unknown.

4.3.2 The consistency values of positive and negative pairs. To demonstrate the superiority of utilizing the learnt reference distribution similarity to generate pseudo pairwise labels, we record the average consistency values (cosine similarities) of real positive pairs and

Table 2: Open set evaluation on the Events2012.

Blocks	M_1		M_2		M_3		M_4		M_5		M_6		M_7	
Metrics	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI
TwitterLDA	.11±.00	.08±.00	.27±.01	.20±.01	.28±.00	.22±.01	.25±.00	.17±.00	.26±.00	.21±.00	.32±.00	.20±.00	.18±.01	.12±.01
Word2Vec	.19±.00	.08±.00	.50±.00	.41±.00	.39±.00	.31±.00	.34±.00	.24±.00	.41±.00	.33±.00	.53±.00	.40±.00	.25±.00	.13±.00
BERT	.36±.00	.34±.00	.78±.00	.76±.00	.75±.00	.73±.00	.60±.00	.55±.00	.72±.00	71±.00	.78±.00	.74±.00	.54±.00	.50±.00
EventX	.36±.00	06±.00	.68±.00	29±.00	.63±.00	.18±.00	.63±.00	19±.00	.59±.00	14±.00	.70±.00	27±.00	.51±.00	13±.00
PP-GCN	.23±.00	.21±.00	.57±.02	.55±.02	.55±.01	.52±.01	.46±.01	.42±.01	.48±.01	.46±.01	.57±.01	52±.02	.37±.00	.34±.00
KPGNN	.39±.00	.37±.00	.79±.01	.78±.01	.76±.00	.74±.00	.67±.00	.64±.01	.73±.01	.71±.01	.82±.01	.79±.01	.55±.01	.51±.01
QSGNN	**.43±.01**	**.41±.02**	**.81±.02**	**.80±.01**	**.78±.01**	**.76±.01**	**.71±.02**	**.68±.01**	**.75±.00**	**.73±.00**	**.83±.01**	**.80±.01**	**.57±.01**	**.54±.00**
promotion	↑ 4%	↑ 4%	↑ 2%	↑ 2%	↑ 2%	↑ 2%	↑ 3%	↑ 3%	↑ 2%	↑ 2%	↑ 1%	↑ 1%	↑ 2%	↑ 3%

Blocks	M_8		M_9		M_{10}		M_{11}		M_{12}		M_{13}		M_{14}	
Metrics	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI
TwitterLDA	.37±.01	24±.01	.34±.00	.24±.00	.44±.01	.36±.01	.33±.01	.25±.01	.22±.01	.16±.01	.27±.00	.19±.00	.21±.00	.15±.01
Word2Vec	.46±.00	.33±.00	.35±.00	.24±.00	.51±.00	.39±.00	.37±.00	.26±.00	.30±.00	.23±.00	.37±.00	.23±.00	.36±.00	.26±.00
BERT	.79±.00	.75±.00	.70±.00	.66±.00	.74±.00	.70±.00	.68±.00	.65±.00	.59±.00	.56±.00	.63±.00	.59±.00	.64±.00	.61±.00
EventX	71±.00	.21±.00	.67±.00	.19±.00	.68±.00	.24±.00	.65±.00	.24±.00	.61±.00	.16±.00	.58±.00	.16±.00	.57±.00	.14±.00
PP-GCN	.55±.02	.49±.02	.51±.02	.46±.02	.55±.02	.51±.02	.50±.01	.46±.02	.45±.01	.42±.01	.47±.01	.43±.01	.44±.01	.41±.01
KPGNN	**.80±.00**	**76±.01**	.74±.02	.71±.02	.80±.01	78±.01	.74±.01	.71±.01	.68±.01	66±.01	**.69±.01**	**.67±.01**	**.69±.00**	.65±.00
QSGNN	.79±.01	.75±.01	**.77±.02**	**.75±.02**	**.82±.02**	**.80±.03**	**.75±.01**	**.72±.01**	**.70±.00**	**.68±.00**	.68±.02	.66±.01	.68±.01	**.66±.01**
promotion	↓ 1%	↓ 1%	↑ 3%	↑ 4%	↑ 2%	↑ 2%	↑ 1%	↑ 1%	↑ 2%	↑ 2%	↓ 1%	↓ 1%	↓ 1%	↑ 1%

Blocks	M_{15}		M_{16}		M_{17}		M_{18}		M_{19}		M_{20}		M_{21}	
Metrics	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI
TwitterLDA	.21±.00	.13±.00	.35±.01	.27±.01	.19±.00	.13±.00	.18±.00	.12±.00	.29±.01	.22±.00	.35±.00	.23±.00	.19±.00	.13±.00
Word2Vec	.27±.00	.15±.00	.49±.00	.36±.00	.33±.00	.24±.00	.29±.00	.21±.00	.37±.00	.28±.00	.38±.00	.24±.00	.31±.00	.21±.00
BERT	54±.00	.50±.00	.75±.00	.72±.00	.63±.00	.60±.00	.57±.00	.53±.00	.66±.00	.63±.00	.68±.00	.62±.00	.59±.00	57±.00
EventX	.49±.00	07±.00	.62±.00	.19±.00	.58±.00	.18±.00	.59±.00	.16±.00	.60±.00	.16±.00	.67±.00	.18±.00	.53±.00	.10±.00
PP-GCN	.39±.01	.35±.01	.55±.01	.52±.01	.48±.00	.45±.00	.47±.01	.45±.01	.51±.02	.48±.02	.51±.01	.45±.02	.41±.02	.38±.02
KPGNN	.58±.00	.54±.00	**.79±.01**	**.77±.01**	.70±.01	.68±.01	.68±.02	.66±.02	**.73±.01**	**.71±.01**	.72±.02	.68±.02	**.60±.00**	.57±.00
QSGNN	**.59±.01**	**.55±.01**	.78±.01	.76±.02	**.71±.01**	**.69±.01**	**.70±.01**	**.68±.01**	**.73±.00**	.70±.01	**.73±.02**	**.69±.02**	**.61±.01**	**.58±.00**
promotion	↑ 1%	↑ 1%	↓ 1%	↓ 1%	↑ 1%	↑ 1%	↑ 2%	↑ 2%	-	↓ 1%	↑ 1%	↑ 1%	↑ 1%	↑ 1%

Table 3: Open set evaluation on the Events2018.

Blocks	M_1		M_2		M_3		M_4		M_5		M_6		M_7		M_8	
Metrics	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI
TwitterLDA	.20±.00	19±.00	.09±.00	06±.00	.13±.00	.11±.00	.10±.00	.08±.00	.24±.00	.20±.00	.22±.00	.19±.00	.12±.00	.10±.00	.24±.00	.20±.00
Word2Vec	.22±.00	21±.00	.22±.00	21±.00	.25±.00	23±.00	.28±.00	27±.00	.48±.00	46±.00	.33±.00	31±.00	.35±.00	.33±.00	.37±.00	34±.00
BERT	.32±.00	.28±.00	.32±.00	.31±.00	.31±.00	.32±.00	.33±.00	.30±.00	.47±.00	.44±.00	.36±.00	.33±.00	.41±.00	.36±.00	.44±.00	.38±.00
EventX	.34±.00	.11±.00	.37±.00	.12±.00	.37±.00	.11±.00	.39±.00	.14±.00	.53±.00	.24±.00	.44±.00	.15±.00	.41±.00	.12±.00	.54±.00	.21±.00
PP-GCN	.49±.01	48±.00	.45±.00	44±.02	.56±.03	.55±.03	.54±.03	.54±.04	.54±.02	.53±.02	.52±.02	.50±.03	.56±.04	.55±.04	.56±.03	.55±.02
KPGNN	.54±.01	.54±.01	.56±.02	.55±.01	.52±.03	.55±.02	.55±.01	.55±.01	.58±.02	57±.01	.59±.03	.57±.02	.63±.02	61±.02	**.58±.02**	**.57±.02**
QSGNN	**.57±.01**	**.56±.01**	**.58±.01**	**.57±.01**	**.57±.01**	**.56±.02**	**.58±.03**	**.57±.03**	**.61±.02**	**.59±.01**	**.60±.01**	**.59±.01**	**.64±.01**	**.63±.01**	.57±.02	.55±.02
promotion	↑ 2%	↑ 1%	↑ 2%	↑ 2%	↑ 1%	↑ 1%	↑ 3%	↑ 2%	↑ 3%	↑ 2%	↑ 1%	↑ 2%	↑ 1%	↑ 2%	↓ 2%	↓ 2%

Blocks	M_9		M_{10}		M_{11}		M_{12}		M_{13}		M_{14}		M_{15}		M_{16}	
Metrics	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI	NMI	AMI
TwitterLDA	.16±.00	.12±.00	.17±.00	.11±.00	.22±.00	.18±.00	.28±.00	.25±.00	.19±.00	.17±.00	.24±.00	.21±.00	.33±.00	.30±.00	.07±.00	.02±.0
Word2Vec	.33±.00	.30±.00	.46±.00	.42±.00	.41±.00	38±.00	.40±.00	.37±.00	.22±.00	20±.00	.36±.00	34±.00	.41±.00	38±.00	.28±.00	.25±.00
BERT	.38±.00	.28±.00	.42±.00	.35±.00	.45±.00	.34±.00	.48±.00	.44±.00	.31±.00	.26±.00	.43±.00	.40±.00	.39±.00	.39±.00	.34±.00	.27±.00
EventX	.45±.00	.16±.00	.52±.00	.19±.00	.48±.00	.18±.00	.51±.00	.20±.00	.44±.00	.15±.00	.52±.00	.22±.00	.49±.00	.22±.00	.39±.00	.10±.00
PP-GCN	**.54±.02**	**.48±.03**	.56±.06	.55±.04	.59±.03	.57±.02	.60±.02	.58±.02	**.61±.01**	.59±.02	.60±.02	.59±.01	.57±.03	.55±.03	**.53±.02**	**.52±.02**
KPGNN	.48±.02	.46±.02	.57±.01	.56±.02	.54±.01	.53±.01	.55±.04	.56±.02	.60±.02	**.60±.02**	.66±.01	.65±.00	.60±.01	.58±.02	.52±.02	.50±.01
QSGNN	.52±.02	.46±.02	**.60±.01**	**.58±.01**	**.60±.01**	**.59±.02**	**.61±.02**	**.59±.02**	.59±.04	.58±.03	**.68±.02**	**.67±.02**	**.63±.02**	**.61±.00**	.51±.03	.50±.03
promotion	↓ 2%	↓ 2%	↑ 3%	↑ 2%	↑ 1%	↑ 2%	↑ 1%	↑ 1%	↓ 2%	↓ 2%	↑ 2%	↑ 2%	↑ 3%	↑ 3%	↓ 2%	↓ 2%

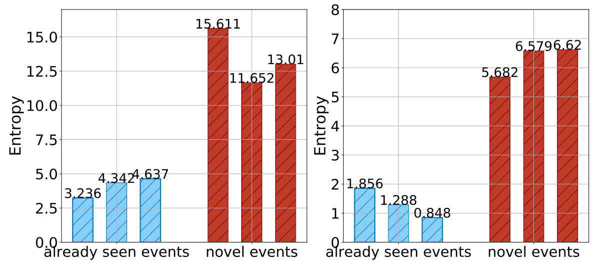
real negative pairs calculated from the initially learnt representations and the RSD vectors, respectively, in Table 4. Obviously, when calculated from the RSD vectors, the consistency values of the positive pairs and the negative pairs differ more. Thus we can give more accurate judgement to the positive/negative pairs. Hence, the proposed RSD vector is helpful.

4.3.3 Using entropy to measure the diversity of events. We plot the average entropy of 3 randomly selected events which are previously seen in the pre-training stage and 3 randomly selected novel events which have not appeared in the pre-training stage in Fig. 5. Apparently, those novel events have higher information entropy which means they are more different from the known data. This validates the effectiveness of using entropy to sample diverse events.

4.3.4 Using consistency value to measure pairwise label quality. We plot the distributions of consistency values of real positive pairs and real negative pairs in Fig. 6. When the consistency value > 0.5, as

Table 4: Consistency values calculated from the initial representations *h* vs. from RSD vectors.

label	Events2012		Events2018	
	Consistency of ***h***	Consistency of RSD	Consistency of ***h***	Consistency of RSD
Positive	0.7287	0.7644	0.6576	0.8350
Negative	0.1834	0.1002	0.3023	0.3459
difference	0.5453	0.6642	0.3553	0.4891



(a) Events2012 (b) Events2018

Figure 5: The diversity of events measured by entropy.

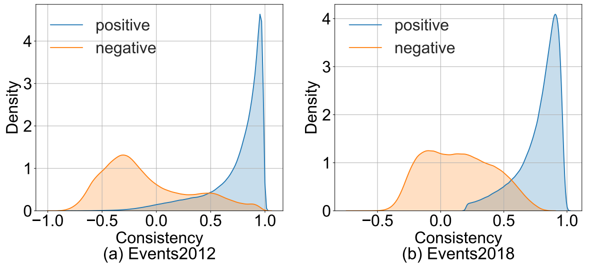
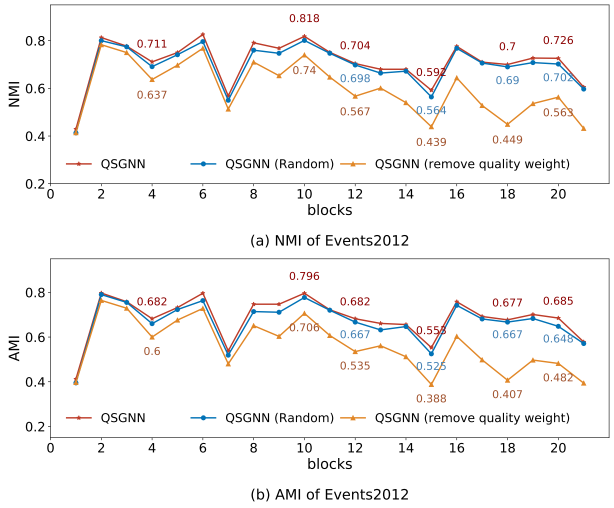


Figure 6: The distribution of consistency values.



(a) NMI of Events2012

(b) AMI of Events2012

Figure 7: Results with different selection strategy and loss.

the value increases, the percentage of positive pairs also increases. Similarly, when the consistency value < 0.5, as the value decreases, the percentage of negative pairs gets higher. When the consistency is at a maximum or minimum, the qualities of the pseudo labels are highest. However, when the consistency value is close to the threshold (0.5), the positive and negative labels have the highest mixing ratio thus are unreliable. Fig. 6 demonstrates the relation between the consistency value and the label distribution to measure the label quality.

4.3.5 Ablation study. To validate the usefulness of (1) the selection strategy based on the diversity principle and (2) the quality-aware optimization, we perform an ablation study. For (1), we remove the unbalanced sampling strategy, as mentioned in Sec. 3.4.2, and adopt a *Random* strategy to make a comparison. Specifically, we randomly select 15 negative and 15 positive pairs of each message. For (2), we remove the quality weight and adopt $\mathcal{L}_t$ to compute the loss. We show the results in Fig. 7. When the unbalanced selection strategy is replaced by the *Random* selection strategy, the performances drop slightly. This demonstrates the superiority of the diversity-based sample selection. By selecting more samples which are different from the known events, the model better adapts to the newly emerging data. Besides, from Fig. 7, we can see that when the quality-aware optimization is removed, the results have a significant decline. This validates the indispensability of the quality assessment. By measuring the quality of pseudo labels and adjusting their contributions to the loss, the fine-tuning is more reliable.

5 CONCLUSION

We have presented a quality-aware self-improving GNN framework to tackle the challenging problem of open set social event detection. First, to make the best of those known events, we extend the conventional triplet loss to a more strict pairwise loss with an orthogonal constraint to train the GNN encoder. Next, to generalize from known to unknown in an effective and reliable way, we propose to use the reference similarity distribution vectors for pseudo pairwise label generation, selection and quality assessment. Specifically, the selection strategy follows the principle of diversity and the quality is measured by consistency. A quality-aware optimization strategy is proposed to resist the noise by re-weighting the contributions of different pseudo labels. Experimental results illustrate that our model achieves state-of-the-art results in both closed set setting and open set setting.

ACKNOWLEDGMENTS

The authors of this paper were supported by the National Key R&D Program of China through grant 2021YFB1714800, NSFC through grants U20B2053 and 62002007, S&T Program of Hebei through grant 21340301D, Beijing Natural Science Foundation through grant 4222030, and the Fundamental Research Funds for the Central Universities. Philip S. Yu was supported by NSF under grants III-1763325, III-1909323, III-2106758, and SaTC-1930941. Thanks for computing infrastructure provided by Huawei MindSpore platform. For any correspondence, please refer to Hao Peng.

REFERENCES

[1] Charu C Aggarwal and Karthik Subbian. 2012. Event detection in social streams. In *Proceedings of the 2012 SIAM international conference on data mining*. SIAM, 624–635.

[2] Hadi Amiri and Hal Daumé III. 2016. Short text representation for detecting churn in microblogs. In *AAAI*. 2566–2572.

[3] Farzindar Atefeh and Wael Khreich. 2015. A survey of techniques for event detection in twitter. *Computational Intelligence* 31, 1 (2015), 132–164.

[4] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. 2018. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 9368–9377.

[5] Mustafa Bilgic and Lise Getoor. 2009. Link-based active learning. In *NIPS Workshop on Analyzing Networks and Learning with Graphs*, Vol. 4.

[6] 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 *The Web conference*.

[7] Wanqiu Cui, Junping Du, Dawei Wang, Feifei Kou, and Zhe Xue. 2021. MVGAN: Multi-View Graph Attention Network for Social Event Detection. *ACM Transactions on Intelligent Systems and Technology (TIST)* 12, 3 (2021), 1–24.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL* (2018).

[9] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, Vol. 96. 226–231.

[10] Mateusz Fedoryszak, Brent Frederick, Vijay Rajaram, and Changtao Zhong. 2019. Real-time event detection on social data streams. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2774–2782.

[11] Wei Feng, Chao Zhang, Wei Zhang, Jiawei Han, Jianyong Wang, Charu Aggarwal, and Jianbin Huang. 2015. STREAMCUBE: Hierarchical spatio-temporal hashtag clustering for event exploration over the Twitter stream. In *2015 IEEE 31st international conference on data engineering*. IEEE, 1561–1572.

[12] Alexander Freytag, Erik Rodner, and Joachim Denzler. 2014. Selecting influential examples: Active learning with expected model output changes. In *European conference on computer vision*. Springer, 562–577.

[13] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *International Conference on Machine Learning*. PMLR, 1183–1192.

[14] Li Gao, Hong Yang, Chuan Zhou, Jia Wu, Shirui Pan, and Yue Hu. 2018. Active discriminative network representation learning. In *IJCAI*. 2142–2148.

[15] Anuradha Goswami and Ajey Kumar. 2016. A survey of event detection techniques in online social networks. *Social Network Analysis and Mining* 6, 1 (2016), 1–25.

[16] Yuhong Guo. 2010. Active instance sampling via matrix partition. In *Proceedings of the 23rd International Conference on Neural Information Processing Systems-Volume 1*. 802–810.

[17] K Han, SA Rebuffi, S Ehrhardt, A Vedaldi, and A Zisserman. 2020. Automatically discovering and learning new visual categories with ranking statistics. In *Proceedings of the 8th Intennational Conference on Learning Representations, ICLR 2020*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[18] Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2019. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8401–8409.

[19] Yen-Chang Hsu, Zhaoyang Lv, Joel Schlosser, Phillip Odom, and Zsolt Kira. 2018. Multi-class classification without multi-class labels. In *International Conference on Learning Representations*.

[20] Xuhui Jia, Kai Han, Yukun Zhu, and Bradley Green. 2021. Joint representation learning and novel category discovery on single-and multi-modal data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 610–619.

[21] Qian Li, Hao Peng, Jianxin Li, Yiming Hei, Rui Sun, Jiawei Sheng, Shu Guo, Lihong Wang, and Philip S. Yu. 2021. Deep Learning Schema-based Event Extraction: Literature Review and Current Trends. *arXiv e-prints* (2021), arXiv–2107.

[22] Bang Liu, Fred X Han, Di Niu, Linglong Kong, Kunfeng Lai, and Yu Xu. 2020. Story forest: Extracting events and telling stories from breaking news. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 14, 3 (2020), 1–28.

[23] Sanmin Liu, Shan Xue, Jia Wu, Chuan Zhou, Jian Yang, Zhao Li, and Jie Cao. 2021. Online active learning for drifting data streams. *IEEE Transactions on Neural Networks and Learning Systems* (2021), 1–15.

[24] Yaopeng Liu, Hao Peng, Jianxin Li, Yangqiu Song, and Xiong Li. 2020. Event detection and evolution in multi-lingual social streams. *Frontiers of Computer Science* 14, 5 (2020), 1–15.

[25] Béatrice Mazoyer, Julia Cagé, Nicolas Hervé, and Céline Hudelot. 2020. A french corpus for event detection on twitter. In *LREC*. 6220–6227.

[26] Andrew J McMinn, Yashar Moshfeghi, and Joemon M Jose. 2013. Building a large-scale corpus for evaluating event detection on twitter. In *KMIS*. 409–418.

[27] Keval Morabia, Neti Lalita Bhanu Murthy, Aruna Malapati, and Surender Samant. 2019. SEDTWik: Segmentation-based event detection from tweets using Wikipedia. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*. 77–85.

[28] Hieu T Nguyen and Arnold Smeulders. 2004. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*. 79.

[29] Ozer Ozdikis, Pinar Karagoz, and Halit Oğuztüzün. 2017. Incremental clustering with vector expansion for online event detection in microblogs. *Social Network Analysis and Mining* 7, 1 (2017), 1–17.

[30] Hao Peng, Jianxin Li, Qiran Gong, Yangqiu Song, Yuanxing Ning, Kunfeng Lai, and Philip S. Yu. 2019. Fine-grained Event Categorization With Heterogeneous Graph Convolutional Networks. In *IJCAI International Joint Conference on Artificial Intelligence*. 3238.

[31] Hao Peng, Jianxin Li, Yangqiu Song, Renyu Yang, Rajiv Ranjan, Philip S. Yu, and Lifang He. 2021. Streaming social event detection and evolution discovery in heterogeneous information networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 5 (2021), 1–33.

[32] Hao Peng, Ruitong Zhang, Shaoning Li, Yuwei Cao, Shirui Pan, and Philip S. Yu. 2022. Reinforced, Incremental and Cross-lingual Event Detection From Social Messages. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).

[33] Hiranmayi Ranganathan, Hemanth Venkateswara, Shayok Chakraborty, and Sethuraman Panchanathan. 2017. Deep active learning for image classification. In *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 3934–3938.

[34] Jiaqian Ren, Lei Jiang, Hao Peng, Zhiwei Liu, Jia Wu, and Philip S. Yu. 2022. Evidential Temporal-aware Graph-based Social Event Detection via Dempster-Shafer Theory. *IEEE ICWS* (2022).

[35] Jiaqian Ren, Hao Peng, Lei Jiang, Jia Wu, Yongxin Tong, Lihong Wang, Xu Bai, Bo Wang, and Qiang Yang. 2021. Transferring Knowledge Distillation for Multilingual Social Event Detection. *arXiv preprint arXiv:2108.03084* (2021).

[36] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. 2021. A survey of deep active learning. *ACM Computing Surveys (CSUR)* 54, 9 (2021), 1–40.

[37] Nicholas Roy and Andrew McCallum. 2001. Toward optimal active learning through monte carlo estimation of error reduction. *ICML, Williamstown* 2 (2001), 441–448.

[38] Sihem Sahnoun, Samir Elloumi, and Sadok Ben Yahia. 2020. Event detection based on open information extraction and ontology. *Journal of Information and Telecommunication* 4, 3 (2020), 383–403.

[39] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 815–823.

[40] Burr Settles, Mark Craven, and Soumya Ray. 2007. Multiple-instance active learning. *Advances in neural information processing systems* 20 (2007).

[41] Greg Corrado Tomas Mikolov, Kai Chen and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of ICLR*.

[42] Simon Tong and Daphne Koller. 2001. Support vector machine active learning with applications to text classification. *Journal of machine learning research* 2, Nov (2001), 45–66.

[43] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. (2018).

[44] Yuan Wang, Jie Liu, Yalou Huang, and Xia Feng. 2016. Using hashtag graph-based topic model to connect semantically-related words without co-occurrence in microblogs. *IEEE Transactions on Knowledge and Data Engineering* 28, 7 (2016), 1919–1933.

[45] Zhongqing Wang and Yue Zhang. 2017. A neural model for joint event detection and summarization. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 4158–4164.

[46] Tsung-Han Wu, Yueh-Cheng Liu, Yu-Kai Huang, Hsin-Ying Lee, Hung-Ting Su, Ping-Chia Huang, and Winston H Hsu. 2021. Redal: Region-based and diversity-aware active learning for point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15510–15519.

[47] Wei Xie, Feida Zhu, Jing Jiang, Ee-Peng Lim, and Ke Wang. 2016. Topicsketch: Real-time bursty topic detection from twitter. *IEEE Transactions on Knowledge and Data Engineering* 28, 8 (2016), 2216–2229.

[48] Chen Xing, Yuan Wang, Jie Liu, Yalou Huang, and Wei-Ying Ma. 2016. Hashtag-based sub-event discovery using mutually generative lda in twitter. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

[49] Xiaohui Yan, Jiafeng Guo, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2015. A probabilistic model for bursty topic discovery in microblogs. In *Twenty-ninth AAAI conference on artificial intelligence*.

[50] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. 2011. Comparing twitter and traditional media using topic models. In *European conference on information retrieval*. Springer, 338–349.

[51] Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. 2021. Neighborhood Contrastive Learning for Novel Class Discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10867–10875.

[52] Zhun Zhong, Linchao Zhu, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. 2021. Openmix: Reviving known knowledge for discovering novel visual categories in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9462–9470.