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

Jiaqian Ren1,2, Lei Jiang2, Hao Peng3, Yuwei Cao4, Jia Wu5, Philip S. Yu4, Lifang He6
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

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.
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\footnote{https://github.com/RingBDStack/open-set-social-event-detection}. 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 use 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
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 unlabeled 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 $S = M_0, M_1, \ldots, M_{i-1}, M_i$, is actually a temporal sequence of blocks of social messages, where each message block $M_j, j \in \{0, 1, \ldots, i\}$ 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, \( f(\theta_0, \theta_i) \), that extracts discriminative features and detects events. \( \theta_0 \) denotes the parameter of \( f(\theta_0, \theta_i) \) 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 \( K \) blocks, \( M_0 \) \( M_{k-1} \), which 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, \( f(\theta_0, \theta_i) \), that extracts discriminative features and detects events. \( \theta_0 \) denotes the parameter of \( f(\theta_0, \theta_i) \) 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.

3.3 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 \( \texttt{[41]} \) of its words. The 2-d temporal feature is obtained by converting the timestamp to the OLE date. For each message block, we construct a new graph. We directly inputs the message block into the model to extract a set of features. As mentioned in Sec. 3.1, we split the whole dataset into \( K \) blocks, \( M_0 \) \( M_{k-1} \), which 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, \( f(\theta_0, \theta_i) \), that extracts discriminative features and detects events. \( \theta_0 \) denotes the parameter of \( f(\theta_0, \theta_i) \) 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.

3.3.2 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

\[
L_l = \sum_{(m_o, m_r, m_i) \in T} \max \{ D(h_{m_o}, h_{m_r}) - D(h_{m_r}, h_{m_i}) + a, 0 \},
\]

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 ones that are categorized based on the ground-truth events labels and demand the cosine similarity of the learnt message representations be close to it. Suppose \( m_1 \) and \( m_2 \) belong to different classes thus improves the events clustering results.

where \( \{\text{Pos}\} \) denotes the positive pairs and \( \{\text{Neg}\} \) denotes the negative ones. Obviously, with the help of this loss, the minimum intra-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 converge 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 \( P \in \mathbb{R}^{N_b \times N_b} \). \( P_{ij} = 1 \) 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 \( \mathbf{H} \cdot \mathbf{H}^T \), where \( \mathbf{H} \in \mathbb{R}^{N_b \times d} \) denotes the normalized message representations. The additional orthogonal loss is:

\[
L_o = \text{Sum}((P - \mathbf{H}) \times \mathbf{H}^T)^2,
\]

where \( \text{Sum}(\cdot) \) 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 \( \mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_K] \in \mathbb{R}^{K \times d} \) denotes a matrix which stores the normalized representations of \( K \) known reference events, with each row denoting a reference event. For example, \( \mathbf{r}_{ek} \in \mathbb{R}^d \) 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 \( \mathbf{h}_m \), its RSD vector \( \mathbf{p}_{m_i} \in \mathbb{R}^K \) is calculated as:

\[
\mathbf{p}_{m_i} = \text{Softmax}(\mathbf{h}_m, \mathbf{R}^T).
\]

The RSD vector, \( \mathbf{p}_{m_i} \), captures the global similarities between \( m_i \) and those known events. It contains much more knowledge compared to \( \mathbf{h}_m \), 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:

\[
C(m_i, m_j) = \frac{\mathbf{p}_{m_i} \cdot (\mathbf{p}_{m_j})^T}{||\mathbf{p}_{m_i}||_2 \cdot ||(\mathbf{p}_{m_j})^T||_2},
\]

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 $p_{m_i}$, the entropy is:

$$H(p_{m_i}) = -\sum_{j=1}^{K} p_{m_ij} \log_2(p_{m_ij}).$$  \hfill (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, $C$ (Pos), of those pseudo positive pairs ($\text{consistency} > 0.5$) as the quality, and assign $1 - C(\text{Neg})$ 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:

$$\mathcal{L}_{qp} = \sum_{(m_i, m_j^+) \in \text{(Pos)}} (C(m_i, m_j^+) + 1 - C(m_j, m_j^-)) \cdot \max\{D(h_{m_i}, h_{m_j^+}) - D(h_{m_i}, h_{m_j^-}) + a, 0\}. \hfill (8)$$

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
We randomly sample 20% of the initial block for testing, 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 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 use the rest 70% for training.

4.3.1 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 Lo) 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 representations learnt by (QSGNN w/o Lo) to be more distinguishable from the inter-class ones. Meanwhile, the observations of intra-class representations learnt by (QSGNN w/o Lo) 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 Lo) demonstrates the effects of adding orthogonal constraint during training.

4.3.2 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 Lo) 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 Lo) demonstrates the effects of adding orthogonal constraint during training.
Table 2: Open set evaluation on the Events2012.

<table>
<thead>
<tr>
<th>Blocks</th>
<th>M1</th>
<th>M2</th>
<th>M3</th>
<th>M4</th>
<th>M5</th>
<th>M6</th>
<th>M7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics</td>
<td>NMI</td>
<td>AMI</td>
<td>NMI</td>
<td>AMI</td>
<td>NMI</td>
<td>AMI</td>
<td>NMI</td>
</tr>
<tr>
<td>TwitterLDA</td>
<td>0.12</td>
<td>0.05</td>
<td>0.23</td>
<td>0.01</td>
<td>0.29</td>
<td>0.01</td>
<td>0.26</td>
</tr>
<tr>
<td>Word2Vec</td>
<td>0.19</td>
<td>0.04</td>
<td>0.30</td>
<td>0.01</td>
<td>0.35</td>
<td>0.01</td>
<td>0.31</td>
</tr>
<tr>
<td>BERT</td>
<td>0.36</td>
<td>0.04</td>
<td>0.76</td>
<td>0.01</td>
<td>0.75</td>
<td>0.01</td>
<td>0.60</td>
</tr>
<tr>
<td>EventX</td>
<td>0.36</td>
<td>0.06</td>
<td>0.79</td>
<td>0.01</td>
<td>0.80</td>
<td>0.01</td>
<td>0.76</td>
</tr>
<tr>
<td>PP-GCN</td>
<td>0.25</td>
<td>0.01</td>
<td>0.56</td>
<td>0.01</td>
<td>0.56</td>
<td>0.01</td>
<td>0.46</td>
</tr>
<tr>
<td>KPGNN</td>
<td>0.39</td>
<td>0.07</td>
<td>0.79</td>
<td>0.01</td>
<td>0.76</td>
<td>0.01</td>
<td>0.67</td>
</tr>
<tr>
<td>QSGNN</td>
<td>0.35</td>
<td>0.01</td>
<td>0.31</td>
<td>0.01</td>
<td>0.31</td>
<td>0.01</td>
<td>0.31</td>
</tr>
</tbody>
</table>

promotion: M1 < 4%, M2 < 2%, M3 < 1%, M4 < 1%, M5 < 1%, M6 < 1%, M7 < 1%

Table 3: Open set evaluation on the Events2018.

<table>
<thead>
<tr>
<th>Blocks</th>
<th>M15</th>
<th>M16</th>
<th>M17</th>
<th>M18</th>
<th>M19</th>
<th>M20</th>
<th>M21</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metrics</td>
<td>NMI</td>
<td>AMI</td>
<td>NMI</td>
<td>AMI</td>
<td>NMI</td>
<td>AMI</td>
<td>NMI</td>
</tr>
<tr>
<td>TwitterLDA</td>
<td>0.21</td>
<td>0.13</td>
<td>0.35</td>
<td>0.21</td>
<td>0.19</td>
<td>0.15</td>
<td>0.29</td>
</tr>
<tr>
<td>Word2Vec</td>
<td>0.27</td>
<td>0.15</td>
<td>0.49</td>
<td>0.26</td>
<td>0.33</td>
<td>0.24</td>
<td>0.29</td>
</tr>
<tr>
<td>BERT</td>
<td>0.54</td>
<td>0.20</td>
<td>0.75</td>
<td>0.22</td>
<td>0.63</td>
<td>0.20</td>
<td>0.57</td>
</tr>
<tr>
<td>EventX</td>
<td>0.49</td>
<td>0.07</td>
<td>0.62</td>
<td>0.01</td>
<td>0.58</td>
<td>0.18</td>
<td>0.59</td>
</tr>
<tr>
<td>PP-GCN</td>
<td>0.39</td>
<td>0.01</td>
<td>0.51</td>
<td>0.01</td>
<td>0.48</td>
<td>0.01</td>
<td>0.47</td>
</tr>
<tr>
<td>KPGNN</td>
<td>0.58</td>
<td>0.04</td>
<td>0.79</td>
<td>0.01</td>
<td>0.70</td>
<td>0.01</td>
<td>0.68</td>
</tr>
<tr>
<td>QSGNN</td>
<td>0.59</td>
<td>0.01</td>
<td>0.55</td>
<td>0.01</td>
<td>0.71</td>
<td>0.01</td>
<td>0.70</td>
</tr>
</tbody>
</table>

promotion: M15 < 1%, M16 < 1%, M17 < 1%, M18 < 2%, M19 < 2%, M20 < 1%, M21 < 1%

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
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 $L_{1,2}$ 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


