Information Sciences 609 (2022) 1030–1051

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A weakly-supervised graph-based joint sentiment topic model for multi-topic sentiment analysis

Tao Zhou [a], Kris Law [a,*], Douglas Creighton [b]

[a] School of Engineering, Faculty of Science Engineering and Built Environment, Deakin University, Geelong, VIC 3217, Australia
[b] Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Geelong, VIC 3217, Australia

ARTICLE INFO

Article history:
Received 15 December 2021
Received in revised form 9 July 2022
Accepted 23 July 2022
Available online 27 July 2022

Keywords:
Multi-topic sentiment analysis
Edge-gated graph convolutional network
Joint sentiment-topic model
Self-training

ABSTRACT

Multi-topic sentiment analysis, which aims to identify the topics and classify their corresponding sentiment, is of great value in understanding consumers' behaviour and improving services. Because of the high cost of manual annotation of the datasets, topic model-based approaches that model the joint distributions of both topics and sentiments have been studied previously. Some studies proposed models that leverage the prior knowledge derived from the pre-trained word embeddings and have proven effective. However, most of the existing models are based on the assumption that words and topics are conditionally independent, ignoring the dependency relations among them. Additionally, the fine-tuning of the pre-trained word embeddings to incorporate the contextual information is also neglected in these models. This could result in the ambiguous representations of topics. In this paper, we propose a novel weakly-supervised graph-based joint sentiment topic model (W-GJST) that integrates an edge-gated graph convolutional network (E-GCN) into a joint sentiment-topic model. An importance sampling-based training method is proposed to learn the contextual representations of topics and words efficiently. Additionally, a self-training multi-topic classifier is designed for the multi-label topic identification. Experiments on two benchmark datasets demonstrate the superiority of the proposed W-GJST compared to the baseline models in terms of topic modelling, topic identification and topic-sentiment identification.

1. Introduction

With the rapid developments in digitalisation and the ever-growing popularity of mobile devices, an explosive growth of user-generated textual data has been witnessed. For example, consumers increasingly tend to leave online comments and feedback on social media and e-commerce platforms, such as Facebook, LinkedIn and Amazon, to express their opinions towards the products or services received. By analysing and mining online comments, organisations could better understand the preferences and interests of consumers and, therefore, improve their services, products and marketing campaigns accordingly. Rather than expressing their overall satisfaction, customers frequently comment on their experience of purchasing or using a product from different aspects or topics. Likewise, organisations may also pre-define a number of topics for the comments they are interested in analysing. Therefore, automatically identifying which topics are mentioned in reviews becomes a valuable and fundamental task for applications, such as user profiling [1]. Meanwhile, the sentiment opinions

* Corresponding author.
E-mail address: kris.law@deakin.edu.au (K. Law).

https://doi.org/10.1016/j.ins.2022.07.126
0020-0255/© 2022 Elsevier Inc. All rights reserved.

of customers towards these topics are also vital feedback to evaluate the performance of products or services. Traditionally, sentiment analysis gives a rough sentiment polarity of a review, such as "positive", "negative", "satisfied" or "unsatisfied", which is insufficient to provide a thorough profile of consumers' preferences. A fine-grained sentiment analysis that shows the sentiment opinions towards each identified topic enables building a more detailed profile of consumer preferences. We take a restaurant review as an example: "*my son really enjoys the place, great snacks such as chips and chicken wings, and causal kid friendly playground with a wide range of toys.*". Firstly, we can identify two topics in this review, namely *food* and *child-care facility*. These two topics are not explicitly mentioned in the review, but they can be inferred through some keywords. Topic *food* can be implied by words "snacks", "chips" and "chicken wings", while the topic *child-care facility* is indicated by words "kid", "playground" and "toys". Other customers may also comment on the topics such as *service* and *price*, which are not discussed in this review. All these topics can be pre-defined and the process of detecting which topics are discussed in the reviews can be referred as topic identification. For the topics *food* and *child-care facility*, the consumer expresses their opinion with "great" and "friendly". Hence, we could identify both of their corresponding sentiments as "positive". We refer to the process of identifying the associations among topics and sentiments as the topic-sentiment identification. In this paper, we aim to deal with such a fine-grained sentiment analysis problem, denoted by the multi-topic sentiment analysis, which aims to identify multiple topics covered by each document (i.e. topic identification) and their corresponding sentiment labels (i.e. topic-sentiment identification) simultaneously.

In recent years, studies on topic identification and sentiment analysis have flourished. Deep learning-based approaches have achieved outstanding performance on the aspect-based sentiment analysis (ABSA) in many studies. Most deep learning-approaches deal with ABSA by building an end-to-end multi-class sentiment classifier in a supervised model with various neural networks, such as CNN [2] and attention-based networks [3]. Although a significant improvement of the performance on ABSA has been observed by adopting supervised deep learning-based methods, the lack of labelled data prohibits the implementation in many scenarios. Extracting topics and sentiments manually is highly labour-intensive and time-consuming. Most users cannot afford to manually annotate the extensive amount of training data, especially when domain knowledge is needed. Additionally, it is challenging for deep learning approaches to extract the associations among topics, sentiments and words. In practice, organisations are keen to understand the features of a topic discussed the most by consumers. For instance, what features of topic *food* lead to a favourable opinion? In the example discussed earlier, the feature triggering positive opinion is "snacks". Therefore, the interpretability of deep learning approaches is limited. Unsupervised or weakly supervised topic model-based approaches, which usually do not need labelled data, have been studied to cope with these drawbacks. One of the most representative unsupervised methods for topic modelling, Latent Dirichlet Allocation (LDA) [4] is a generative model, which assumes that each topic is represented by a distribution of words. This assumption reduces the data dimensionality, and LDA is effective on long documents [5]. Due to the success of LDA in topic modelling, some studies have extended LDA to joint models for dealing with both topic identification and topic-sentiment analysis. The joint sentiment-topic model (JST) [6] extends the framework of LDA by adding a sentiment layer to generate a probability distribution of sentiment labels. Similarly, the unsupervised topic-sentiment joint probabilistic model (UTSJ) [7] assumes that the expressed sentiment should depend on the specific topic. Although these models can obtain joint topic-sentiment distributions, they may fail to extract the semantic associations between low-frequency words with topics and sentiment labels [8]. The problem becomes severe for short-text documents with data sparsity and seriously impedes the process of generating satisfactory joint distributions [4]. Some studies propose considering external resources to enhance the semantic understanding of words and topics to mitigate the data sparsity problem. Word embeddings pre-trained on large corpus have been widely applied in various natural language processing tasks, significantly improving performance [9]. W2VLDA [10] combines the LDA with the pre-trained word embeddings and a maximum entropy classifier to perform on ABSA in an unsupervised mode. W2VLDA initialises a group of seed words for each topic and sentiment label and constructs biased hyper-parameters of LDA based on embedding similarity. Fu et al. (2018) [11] proposed a novel weakly supervised topic sentiment model (WS-TSWE) that extends JST by incorporating the embedding vectors of topics for topic-word relation exploration and HowNet lexicon for sentiment recognition. WS-TSWE updates the topic representations through maximum likelihood estimation and defines a multinomial distribution of generating a word from embedding components to inject the word embedding information into the model.

Despite the successful applications of these topic model-based approaches in topic identification and sentiment analysis, open challenges remain. Firstly, the LDA and its extensions are developed based on the assumption that the words are conditionally independent. The topic distributions and topics are represented as multinomial distributions of words without considering their dependency relation [12]. Ignoring the fact that words in a corpus are often dependent on each other linguistically and contextually, however, could lead to ambiguous representation of a topic. For example, the identification of the two words "deep" and "learning" for a topic is far more meaningful than only discovering the word "learning". Secondly, even though the pre-trained word embeddings could bring prior knowledge to the joint models, the current approaches, such as WS-TSWE, learn the topic representations iteratively with constant word embeddings. The word embeddings could be fine-tuned along with the training of topic representations, however, to better incorporate contextual information. Additionally, in WS-TSWE, the learning of topic representations requires processing the whole vocabulary of documents, which could become extremely time-consuming when the vocabulary set is large. Thirdly, most topic model-based models are limited to document-level sentiment analysis and are not capable of identifying which topics are discussed in a document. More specifically, the sub-task of topic identification can be formulated as a multi-label classification problem. The existing topic model-based approaches can only output the conditional distributions of topics given a document, which consequentially requires additional steps to identify the existence of a particular topic. A more intuitive method of multi-label topic identification is to set a probability threshold where any topic with a higher probability than the threshold is assumed to be detected, however, selecting an appropriate threshold is rather challenging. Ozyurt and Akcayol (2021) [5] proposed to split sentences into segments such that each segment contains only one topic. Nevertheless, this method requires domain knowledge to pre-define a group of terms related to each topic.

This paper proposes a novel Weakly-supervised Graph-based Joint Sentiment Topic model (W-GJST) for the multi-topic sentiment analysis to address the challenges discussed above. Specifically, as shown in Fig. 1, the proposed W-GJST contains two main modules, the Graph-based JST (GJST) and the Self-training Multi-topic Classifier. The GJST adopts an Edge-gated Graph Convolutional Network (E-GCN) to build a graph representation of topics and words, enabling the model to explore their implicit dependency relations. To overcome the limitation of using constant pre-trained word embeddings in W2VLDA and WS-TSWE, the E-GCN updates the topic and word embeddings by iteratively approximating the probability of the relatedness between words and topics to their empirical distributions. The GJST integrates the E-GCN and the joint sentiment topic model with an embedding component based on the learned topic and word representations. We unify the training process of the E-GCN and the joint sentiment topic model through the Gibbs sampling process. We develop a multi-topic classifier with a self-training scheme to explicitly perform the multi-label topic identification with little supervision. The proposed self-training scheme is a weakly supervised method aiming to train a binary classifier for each topic pseudo-labels generated based on the confidence level of samples and generalises the classifier with a target distribution. Compared to other topic model-based models, the key contributions of this paper can be summarised as follows:

- We propose a novel graph-based joint sentiment topic model (GJST) integrating an Edge-gated Graph Convolutional Network (E-GCN) and a joint sentiment topic model. The E-GCN not only leverages the pre-trained word embeddings to alleviate the data sparsity problem, but also explores the associations among words and topics.
- We propose an importance sampling-based training scheme for the E-GCN to greatly reduces the computational and memory requirements for large documents. The training processes of the E-GCN and the joint sentiment topic model are unified through the Gibbs sampling to discover the co-occurrence information.
- We propose a multi-label topic classifier with an unsupervised self-training scheme, which does not need labelled data and pre-defined terms of topics. The self-training scheme leverages the topic and word embeddings and the conditional distributions of topics obtained from GJST to train a neural network with unlabelled data.
- The proposed weakly-supervised graph-based joint sentiment topic model (W-GJST) is evaluated by comparing the topic model-based methods on two datasets for the multi-topic sentiment analysis. Experimental results demonstrate the effectiveness and superiority of our model in terms of topic identification and topic-sentiment identification.

The remainder of this paper is organised as follows. Section 2 discusses the studies related to joint topic sentiment models and graph representation learning. Section 3 details the proposed W-GJST model framework, including model representation, graph representation learning, parameter estimation and multi-topic identification. Section 4 discusses the experiment settings and results. Finally, Section 5 presents the conclusion and future works.

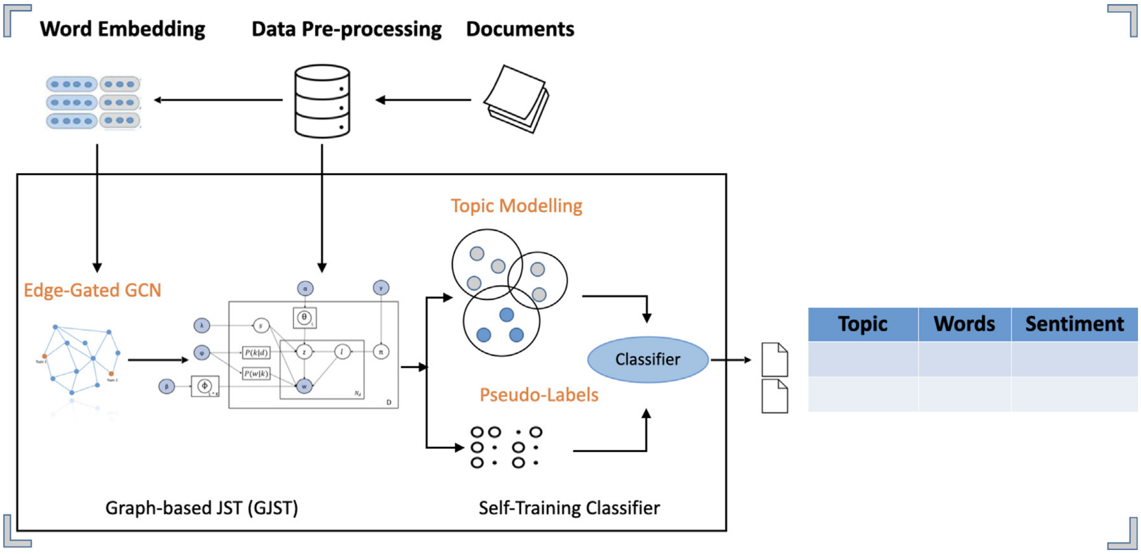


Fig. 1. Framework of Weakly-supervised Graph-based Joint Sentiment-Topic Model.

2. Related Studies

2.1. Topic Model-based Sentiment Analysis

Conventional topic models such as LDA and probabilistic latent semantic analysis (PLSA) [13] extract the topical information and generate the posterior topic-word distribution based on the document-level co-occurrence of words. They assume that documents can be defined by a distribution of topics in which each topic is a distribution of words. Because of the ability to represent the latent features of documents, topic models, especially LDA, have been applied in sentiment analysis. This section reviews the studies on the unsupervised and weakly supervised joint topic sentiment models.

The topic sentiment mixture model (TSM) [14] is a probabilistic model that simultaneously captures the mixture of sentiments and labels. TSM infers separate models for sentiment and topics and utilises a hidden Markov model (HMM) structure to extract the topic and sentiment dynamics. However, TSM does not model the explicit associations among the sentiments and topics [8]. To jointly detect both the topics and sentiments from texts, Lin et al. (2009) [6] proposed a weakly supervised Bayesian model, i.e. JST, which constructs an additional sentiment layer conditioning on the topic layer of LDA. The authors also introduced another Reverse-JST model (RJST), which assumes that sentiments are generated based on the topics [15]. JST is then extended to an aspect sentiment unification model (ASUM) for the online review sentiment analysis [16]. However, ASUM assumes that all words in a single sentence are generated from one topic. Even though both JST and ASUM use external sentiment seed words to guide the generation of the sentiment distribution, JST only uses these words as the prior knowledge of sentiments. At the same time, ASUM integrates the words with the Dirichlet hyper-parameter [17]. Li et al. (2010) [18] proposed a sentiment-LDA model and a dependency-sentiment-LDA model considering both the associations among global topics and sentiments and the local dependency between sentiments. Most recently, Rahman et al. (2016) [19] developed a hidden topic sentiment model (HTSM), which integrates HMM into topic modelling to capture topic coherence and sentiment consistency. Xiong et al. (2018) [20] proposed a word-pair sentiment-topic model (WSTM), which uses a sliding window to generate word pairs. Tang et al. (2019) [8] presented a joint aspect-based sentiment topic (JABST) model, which jointly captures multi-grain aspects and opinions. Besides the prior knowledge of sentiments, word embeddings have been proven effective in improving model performance on both topic modelling and sentiment analysis [11]. However, to our knowledge, few studies have been conducted to integrate the word embeddings with joint topic sentiment models. WS-TSWE [11] is an extension of JST by injecting the word embeddings to learn the topic embeddings. In WS-TSWE, a multinomial probability distribution based on topic embeddings is created to generate words from the embedding component. Sengupta et al. (2021) [21] proposed an embedding-enhanced labelled joint sentiment-topic (LJST) model, which uses Markov random field (MRF) regulariser to create a graph connecting similar words in the documents to improve topic identification. However, LJST is a semi-supervised model that requires labelled documents for training.

2.2. Graph Representation Learning

Graph representation learning has attracted attention over the last few years as the graphical data is ubiquitous in the real world, including knowledge graphs [22] and social networks [23]. Due to their ability to model interactions between entities in a graphical structure, graph-based methods have been widely adopted in natural language processing tasks, such as text classification [24], sentiment analysis [25], and recommendation system [26]. We define graph representation learning as a process of converting the raw graph data into a high-dimensional vector by using graph embedding techniques [27]. There are three main types of graph embedding methods: factorisation-based, random walk-based, and deep learning-based.

Factorisation-based methods formulate the connections between two nodes on a graph as a matrix and factorise the matrix to obtain the embedding vectors [28]. For example, the graph factorisation (GF) algorithm factorises the adjacency matrix of the graph by minimising the distance between the edge features and the inner product of attributes of the vertexes connecting this edge [29]. Factorisation-based algorithms, however, are not scalable or computationally efficient for large graphs. To deal with large graphs, random walk-based methods explore the structural information of graphs by sampling a large number of paths starting from random vertexes [28]. DeepWalk [30] and Node2Vec [31] are the two most popular random walk-based methods, and both algorithms aim to maximise the likelihood of observing the neighbourhood of a node in random walks. The key difference is that Node2Vec adopts a breadth-first sampling (BFS) strategy rather than depth-first sampling (DFS) in DeepWalk. With the ability to preserve high-order proximity and approximating complex functions, deep learning-based approaches, especially graph neural networks (GNN), are becoming more popular. Graph convolutional network (GCN) uses convolution operators to aggregate the features of neighbouring nodes to update the graph embeddings iteratively [32]. Additionally, to address the scalability issue and deal with large graphs, embedding methods such as graph partition neural network (GPNN) [33] and LINE [34] are studied.

To our knowledge, few studies have been conducted to incorporate graph representation learning with topic model-based approaches. Gao et al. [35] proposed a LDA-based model of the expert topic analysis which incorporates the paper cooperation network. In their model, an experts' paper cooperation network is built through analysing the authors' cooperative relations on papers. This paper cooperation network is then used as a constraint of expert-topic distribution to the LDA model. However, the paper cooperation network is formulated as conditional expert-topic distributions instead of embeddings. Long et al. [36] proposed a graph structural topic neural network (GraphSTONE), which first derives the topic models of graphs through a graph anchor LDA model and then applies a multi-view GCN to aggregate topic features. To integrate the training process of GCN and LDA, a graph convolutional topic model (GCTM) [37] is proposed. GCTM applies a GCN on a knowledge graph to produce graphical word embeddings with dimensions equal to the topic number. The word embeddings are connected with the prior parameters of topic-word distributions through a linear function. Next, both GCN and LDA are trained jointly with a variational expectation maximisation algorithm.

3. Weakly-supervised Graph-based Joint Sentiment Topic Model

This section presents the development of the W-GJST model proposed for the multi-topic sentiment analysis. Firstly, the model representation of graph-based JST (GJST), including the generative process, is introduced. Secondly, the design of E-GCN and the importance sampling-based training scheme are presented. Then the estimation procedure of parameters of GJST based on the Gibbs sampling algorithm is detailed, along with a multi-topic classifier with a self-training scheme for the topic identification task.

3.1. GJST Model Representation

Considering a collection of review documents $\mathscr{D} = \{W_d\}_{d=1}^{D}$, in which each document is represented by a bag of words $W_d = \{w_1, w_2, \dots, w_{N_d}\}$, where D is the number of documents, N_d is the number of unique words in the dth document. Each word w_n is chosen from a vocabulary of $\mathscr{V}$. According to the joint sentiment-topic modelling frameworks [6], it is assumed that each document W_d can be represented by mixtures of independent sentiment labels and topics. In this section, the sentiment label is denoted by $l \in \{1, 2, \dots, L\}$ and the topic label is denoted by $k \in \{1, 2, \dots, K\}$. Following the assumption of prior distributions on sentiments and topics, the sentiment of the document W_d follows a multinomial distribution π_d sampled from a Dirichlet distribution with parameter γ, denoted by $\pi_d \sim Dir(\gamma)$. Conditioned on each sentiment label l in the document W_d, the topic follows a multinomial distribution $\theta_{d,l}$ sampled from a Dirichlet distribution with parameter α, denoted by $\theta_{d,l} \sim Dir(\alpha)$. In addition, for each sentiment-topic pair (l, k), each word is assumed to follow a multinomial distribution $\phi_{l,k}$ sampled from a Dirichlet distribution with parameter β, denoted by $\phi_{l,k} \sim Dir(\beta)$. The notations used in the GJST model is presented in Table 1.

To incorporate rich information from pre-trained word embeddings and explore the correlation structures of latent topics, a graph convolutional network (GCN) is applied to learn the distributed representations of topics and the weighted correlations between topics and words. The embedding of topic k can be represented by $\upsilon_k \in \mathbb{R}^h$ and the embedding of word w_n is represented by $\omega_n \in \mathbb{R}^h$, where h is the dimension of the embedding vector. Additionally, each document d can be represented by the averaged embedding vector of all words contained. The document embedding of d can be denoted by $\mathbf{W}_d = \frac{1}{N_d}\sum_{i \in W_d} \omega_i$. In this case, we can use vector inner-product based methods, such as cosine similarity, to measure the closeness between these embedding vectors. By normalising these similarities or closeness, the conditional multinomial

Table 1
Notations in GJST.

Term	Definition
$\mathscr{D}$	Set of Documents with the size of D
$\mathscr{V}$	Set of Vocabulary
W_d	Bag of words for dth document, $W_d = \{w_1, \dots, w_{N_d}\}$
N_d	The number of words for dth document
k	kth Topic label, $k \in \{1, 2, \dots, K\}$
l	lth Sentiment label, $l \in \{1, 2, \dots, L\}$
π_d	The multinomial distribution of sentiment label for dth document, sampled from Dirichlet distribution parametrised with γ
$\theta_{d,l}$	The multinomial distribution of topic label given lth sentiment label for dth document, sampled from Dirichlet distribution parametrised with α
$\phi_{l,k}$	The multinomial distribution of word given topic k and sentiment l, sampled from Dirichlet distribution parametrised with β
ω_n	The embedding vector for nth word in Vocabulary
υ_k	The embedding vector for kth topic
λ	$\lambda = \{\lambda_1, \lambda_2\}$, the parameters of Bernoulli distributions for balancing the sampling process of topic and word
τ	The learning rate of the graph representation learning
M	The sample size of the Importance Sampling
φ	The general representation of parameters of Edge-gated ConvNet
$N_{d,l}$	The number of times sentiment l associated with document d
$N_{l,k}$	The number of times words is associated with topic k and sentiment l
$N_{d,l,k}$	The number of times a word from document d is assigned to topic k and sentiment l
$N_{l,k,i}$	The number of times word w_i appeared in topic k and sentiment l
$N_{k,i}$	The number of times word w_i is associated with topic k

probability distributions $P(k|d)$ and $P(w_n|k)$ are derived, where $P(k|d)$ represents the conditional probability of topic k given document d and $P(w_n|k)$ is the conditional probability of word w_n given topic k. The details of the learning of graph representation and the derivation of these conditional distributions are presented in Section 3.2.

The GJST model generates a word w_n in document d as shown in Fig. 2. First, for document d, we draw the sentiment distribution π_d, topic distribution $\theta_{d,l}$ and word multinomial distribution $\phi_{l,k}$ from prior Dirichlet distributions parametrised by γ, α and β. For each word w_n in document d, we sequentially draw a sentiment label l_n from π_d. A topic z_n and a word w_n are generated either from their Dirichlet multinomial distributions or the graph embedding-based multinomial distributions, similar to the mixture of Dirichlet component and word embedding component indicated by[11]. In this section, binary indicators $s_{1,n}$ and $s_{2,n}$ generated from Bernoulli distributions λ_1 and λ_2 are used to control the sampling process. The formal definition of the generative process of GJST is described as follow:

1. For each document d, draw a sentiment multinomial distribution $\pi_d \sim Dir(\gamma)$
2. For each sentiment label l under document d, draw a topic multinomial distribution $\theta_{d,l} \sim Dir(\alpha)$
3. For each sentiment-topic pair (l, k), draw a word distribution $\phi_{l,k} \sim Dir(\beta)$
4. For each word w_n in document d:
   - draw a sentiment label $l_n \sim Mul(\pi_d)$
   - draw a binary indicator $s_{1,n} \sim Ber(\lambda_1)$
   - draw a topic $z_i \sim (1 - s_{1,n})Mul(\theta_{d,l}) + s_{1,n}P(k = z_i|d)$
   - draw a binary indicator $s_{2,n} \sim Ber(\lambda_2)$
   - draw a word $w_n \sim (1 - s_{2,n})Mul\left(\phi_{z_i}\right) + s_{2,n}P(w_n|k = z_i)$

3.2. Learning of Graph Network

To explicitly learn the representations of topics and models the relationship between topics and words, a topic-word graph $G = (V, E)$ is built where $V = \left(v_1, \ldots, v_K, \omega_1, \ldots, \omega_{|\mathcal{V}|}\right)$ is the embedding matrix and $E = \{(i,j), i,j \in V\}$ represents the set of edges between nodes. In the section, the weight of edge e_{ij} is represented by the closeness, i.e. cosine similarity, between node embedding vectors:

$$e_{ij} = sim\left(x_i, x_j\right) = \frac{x_i \cdot x_j}{\|x_i\|\|x_j\|} \tag{1}$$

where $x_i \in V$ and $x_j \in V$ represent the embedding vectors for ith and jth nodes in G, and $\|\cdot\|$ represents the Euclidean norm. Initially, the topic embedding vectors $(v_1, \ldots, v_k)$ are randomly generated. The learning procedure of graph representation consists of two main components, namely the Edge-Gated Graph Convolutional Network (E-GCN) and the importance sampling-based training.

E-GCN. Let x_i^c and e_{ij}^c represent the node embedding vector and edge embedding vector generated from the cth layer E-GCN for node i and edge (i,j), the initial inputs of node embedding x_i^0 and edge embedding e_{ij}^0 are defined as $x_i^0 = W_0^1 x_i$

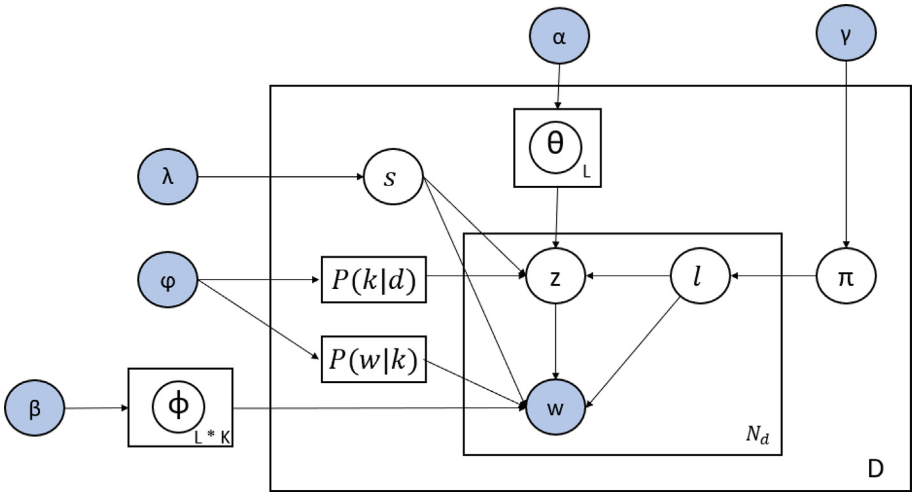


Fig. 2. Graphical representation of GJST Model. The shaded node w denotes the observed words sampled from the prior Dirichlet distribution parametrised by β and the conditional distribution $P(w|k)$ inferred from the E-GCN parametrised by φ.

and $e_{ij}^0 = W_0^2 e_{ij}$, where $W_0^1 \in \mathbb{R}^{h\times h}$ and $W_0^2 \in \mathbb{R}^{h\times 1}$ are linear transformations. The updating procedure of Graph ConvNet proposed by [38] is adopted due to its computational efficiency, as indicated by Eq. 2–4.

$$x_i^{c+1} = x_i^c + ReLU\left(W_1^c x_i^c + \sum_{j\to i} \eta_{ij}^c \odot W_2^c x_j^c\right) \tag{2}$$

$$e_{ij}^{c+1} = e_{ij}^c + ReLU\left(W_3^c e_{ij}^c + W_4^c x_i^c + W_5^c x_j^c\right) \tag{3}$$

$$\eta_{ij}^c = \frac{\sigma\left(e_{ij}^c\right)}{\sum_{j'\to i} \sigma\left(e_{ij'}^c\right) + \varepsilon} \tag{4}$$

where $W \in \mathbb{R}^{h\times h}$, σ is the sigmoid function, ε is a small value, and $ReLU$ is the rectified linear unit. Eq. 2 updates the embedding of node i at $c+1$th layer of Graph ConvNet x_i^{c+1} by aggregating the node embedding at cth layer x_i^c and the weighted sum of neighbouring nodes. η_{ij}^c is an edge-gate that indicates the weighted connectivity between node i and node j, as shown in Eq. 4. By updating the Graph ConvNet for C layers, the probability that a word w_n is generated from the topic v_k can be defined as:

$$P(w_n|k) = \eta_{i=v_k, j=w_n}^C \tag{5}$$

Similarly, we can obtain the embedding of document d by averaging the embedding vectors of the words contained in d. In this case, the probability $P(k|d)$ can be derived as:

$$P(k|d) = \frac{x_{i=v_k}^C \cdot \left(\frac{1}{N_d}\sum_{w_j\in W_d} x_{i=w_j}^C\right)^{\mathrm{T}}}{\sum_k x_{i=v_k}^C \cdot \left(\frac{1}{N_d}\sum_{w_j\in W_d} x_{i=w_j}^C\right)^{\mathrm{T}}} \tag{6}$$

Inspired by the training method in LINE [34], an unsupervised training method for E-GCN is proposed to learn the topic embeddings. The objective of the training is to ensure that $P(w_n|k)$ should be similar to its empirical distribute $\widehat{P}(w_n|k)$. The empirical distribution can be defined as:

$$\widehat{P}(w_n|k) = \frac{f(w_n, k)}{\sum_{(w_i,k)\in E} f(w_i, k)} \tag{7}$$

where $f(w_n, k)$ is the co-occurrence of the word-topic pair (w_n, k). We minimise the KL divergence of $P(\cdot|k)$ and $\widehat{P}(\cdot|k)$ to achieve the objective of training. After omitting the constants, we can have the objective function:

$$\mathscr{L} = min \; - \sum_{(w_n,k)\in E} f(w_n, v_k) P(w_n|k) = min \; - \sum_{(w_n,k)\in E} f(w_n, k)\eta_{v_k,w_n}^C \tag{8}$$

Importance Sampling based training. As indicated by Eq. 2 and Eq. 3, the updates of node embeddings and edge embeddings in the proposed Edge-gated Graph ConvNet require the full expansion of neighbouring nodes and edges. However, the constructed graph becomes rather large-scaled when processing a significant amount of documents with an extensive vocabulary. The training procedure of the Edge-gated Graph ConvNet will be computationally intensive and memory-consuming in this case. Thus, an importance sampling-based training method is proposed. Importance sampling has been widely adopted for graph representation learning to accelerate the training process, alleviate the computational complexity and reduce the variance [39,40]. In the proposed topic-word graph G, the nodes representing the topic embeddings should be kept during the sampling process. Let $\widetilde{A} = \{e_{ij}, i, j \in V\}$ be the edge weight matrix where e_{ij} is the cosine similarity between embedding vectors of node i and j calculated by Eq. 1. As indicated by [39], we can define the sampling probability of word w_n as:

$$q(w_n) = \frac{\|\widetilde{A}(:, w_n)\|^2}{\sum_{w_i\in V/\{v_k\}_{k=1}^K} \|\widetilde{A}(:, w_i)\|^2}, \quad w_n \in \mathscr{V} \tag{9}$$

With the sampling probability, Eq. 2 can be approximated by Monte-Carlo sampling. Assuming that M words are sampling with respect to $q(w_n)$ for each layer of the Edge-gated Graph ConvNet and the set of nodes sampled at cth layer is denoted by s_c, Eq. 2 can be re-written as:

$$x_i^{c+1} = x_i^c + ReLU\left(\frac{1}{M}\sum_{j\in s_c}\frac{A^c(w_i,w_j)}{q(w_j)}x_i^c W^c\right) \tag{10}$$

where $A^c = \left\{\eta_{ij}^c, i,j \in V\right\}$ denotes the edge gate matrix of lth layer and $W^c \in \mathbb{R}^{h\times h}$. Topics nodes are always contained in s_c. The updating of e_{ij}^c and A^c keep unchanged according to Eqs. 3 and 4. Through recursively computing through L layers, the objective function 8 is modified as:

$$\mathscr{L}_s = min \; -\sum_k \sum_{w_n \in s_L} f(w_n,k)\eta_{\upsilon_k,w_n}^C \tag{11}$$

We use the Adam algorithm [26] to optimise Eq. 11 during the training of the GJST model. Algorithm 1 presents the importance sampling-based training process of the Edge-gated Graph ConvNet for one epoch.

Algorithm 1: Importance Sampling based Training (One Epoch)

```
For each word $w_n$, calculate the sampling probability $q(w_n) \propto \|\widetilde{A}(:,w_n)\|^2$ according to Eq. 9
For each layer $c$, sample $M-K$ word related nodes $s_c$ according to $q(w_n), s_c = s_c \cup \{\upsilon_k\}_{k=1}^K$
*Forward Step* ---
**for** For each layer $c$ **do**
  **if** node $\upsilon_i$ is sampled in the next layer, $\upsilon_i \in s_{c+1}$ **then**
    Update the corresponding node embedding $x_i^{c+1}$ of node $\upsilon_i$ according to Eq. 10
    Update the embeddings of edges connecting to $\upsilon_i$ according to Eq. 3 and 4
  **end if**
**end for**
*Optimisation Step* ---
Training Step with Adam optimization algorithm, $W \leftarrow W - \tau \bigtriangledown \mathscr{L}_s$, ($\tau$ - Learning rate)
Update $\widetilde{A}$ according to Eq. 1
```

3.3. Parameter Estimation

The training of the proposed GJST model aims to estimate three sets of latent distribution parameters, namely the document-level sentiment distribution π_d, the sentiment-specific topic distribution $\theta_{d,l}$ and the joint sentiment-topic-word distribution $\phi_{l,k}$. According to the generative process of the GJST model, the joint probability distribution of words, topics and sentiment labels can be decomposed as:

$$p(w,z,l|\alpha,\beta,\gamma,\lambda,\varphi) = p(w|z,l,\alpha,\beta,\gamma,\lambda_2,\varphi)\cdot p(z|l,\lambda_1,\alpha)\cdot p(l|\gamma) \tag{12}$$

where $\lambda = \{\lambda_1,\lambda_2\}$ is the set of parameters of the Bernoulli distributions and φ is a general representation for the parameters of Edge-gated Graph ConvNet. $p(w|z,l,\alpha,\beta,\gamma,\lambda,\varphi)$ is the probability of sampling a word w given a topic z and a sentiment label l, which can be generated by Dirichlet multinomial β and the graph-based multinomial distribution $P(w_n = w|k = z)$. $p(z|l,\alpha)$ is the probability of obtaining a topic z based on a given sentiment label and the prior Dirichlet multinomial distribution α. $p(l|\gamma)$ estimates the sentiment distribution of a document.

For the inference of the probabilistic topic models, several methods have been widely studied, such as Gibbs sampling [11], maximum a posteriori estimation [41] and variational Bayesian inference [1]. This section presents the Gibbs sampling methods for estimating the model parameters due to its good coverage and ease of implementation. Additionally, the training of Edge-gated Graph ConvNet can be easily incorporated with the Gibbs sampling inference of GJST. As the three components in Eq. 12 are assumed independent, they can be computed separately. The first term of Eq. 12 can be calculated by integrating out $\phi_{l,k}$:

$$p(w|z,l,\alpha,\beta,\gamma,\lambda_2,\varphi) = (1-\lambda_2)\left(\frac{\Gamma(|\mathscr{V}|\beta)}{\Gamma(\beta)^{|\mathscr{V}|}}\right)^{L\cdot K}\prod_l\prod_k\frac{\prod_i\Gamma(N_{l,k,i}+\beta)}{\Gamma(N_{l,k}+|\mathscr{V}|\beta)} + \lambda_2 P(w_n = w|k = z,\varphi) \tag{13}$$

where $N_{l,k,i}$ is the number of times word w_i given the topic k and sentiment label $l, N_{l,k}$ is the number of times words assigned to topic k and sentiment label l, $|\mathscr{V}|$ is the size of vocabulary, Γ is the gamma function. Similarly, the second and third terms are calculated by integrating out $\theta_{d,l}$ and π_d:

$$p(z|l,\lambda_1,\alpha) = (1-\lambda_1)\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{D\cdot L}\prod_d\prod_l\frac{\prod_k\Gamma(N_{d,l,k}+\alpha)}{\Gamma(N_{d,l}+K\alpha)} + \lambda_1 P(k = z|d,\varphi) \tag{14}$$

$$p(l|\gamma) = \left(\frac{\Gamma(L\gamma)}{\Gamma(\gamma)^L}\right)^D \prod_d \frac{\prod_l \Gamma(N_{d,l} + \gamma)}{\Gamma(N_d + L\gamma)} \tag{15}$$

where $N_{d,l,k}$ is the number of times a word from document d is assigned to topic k and sentiment label $l, N_{d,l}$ is the number of times sentiment label l associated with document d, N_d is the total number of words in the document d. Gibbs sampling will sequentially sample z_i and l_i from the distribution over z and l given the current values of all other variables and the data. Let the superscript $\neq gi$ denote a quantity that excludes data from the i^{th} position, the conditional posterior distribution for z_i and l_i is:

$$p\left(z_i = k, l_i = l|w, z^{\neg i}, l^{\neg i}, \alpha, \beta, \gamma, \lambda, \varphi\right) \propto \left((1-\lambda_2)\frac{N_{l,k,w_i}^{\neg i} + \beta}{N_{l,k}^{\neg i} + V\beta} + \lambda_2 P(w_i|k, \varphi)\right) \cdot \left((1-\lambda_1)\frac{N_{d,l,k}^{\neg i} + \alpha}{N_{d,l}^{\neg i} + K\alpha} + \lambda_1 P(k|d, \varphi)\right) \cdot \frac{N_{d,l}^{\neg i} + \gamma}{N_d^{\neg i} + L\gamma} \tag{16}$$

Samples derived from the above Markov Chain are then used to estimate π, θ and ϕ, we obtained:

$$\pi_{d,l} = \frac{N_{d,l} + \gamma}{N_d + L\gamma} \tag{17}$$

$$\theta_{d,l,k} = (1-\lambda_1)\frac{N_{d,l,k} + \alpha}{N_{d,l} + K\alpha} + \lambda_1 P(k|d, \varphi) \tag{18}$$

$$\phi_{l,k,i} = (1-\lambda_2)\frac{N_{l,k,i} + \beta}{N_{l,k} + V\beta} + \lambda_2 P(w_i|k, \varphi) \tag{19}$$

The Gibbs sampling procedure of estimating parameters of the GJST model is shown in Algorithm2.

Algorithm2: Gibbs Sampling Procedure of GJST Model Inference

```
Require: Document collection $\mathscr{D} = \{W_d\}_{d=1}^D$, Pre-trained word embeddings $\{\omega_n\}_{n=1}^{|\mathscr{V}|}$, Number of sentiment labels $L$,
    Number of topics $K$, Parameters $\alpha, \beta, \gamma, \lambda_1, \lambda_2$
Initialise $\pi_{d,l}, \theta_{d,l,k}, \phi_{l,k,i}, \varphi$
Initialise topic embeddings $\{\upsilon_k\}_{k=1}^K$ and edge weight matrix $\widetilde{A} = \left\{e_{ij}, i, j \in \mathscr{V} \cup \{\upsilon_k\}_{k=1}^K\right\}$
for document $d = 1$ to $D$ do
  for word $w_i \in W_d$ with $i \in [1, N_d]$ do
    Sample a sentiment label $l \sim \pi_{d,l}$ and a topic $k \sim \theta_{d,l,k}$
    $N_{d,l} \leftarrow N_{d,l} + 1, N_{l,k} \leftarrow N_{l,k} + 1, N_{d,l,k} \leftarrow N_{d,l,k} + 1, N_{l,k,i} \leftarrow N_{l,k,i} + 1, N_{k,i} \leftarrow N_{k,i} + 1$
  end for
end for
for iter  = 1 to maximum iterations do
  Run the Forward Step of Algorithm1
  for document $d = 1$ to $D$ do
    for word $w_i \in W_d$ with $i \in [1, N_d]$ do
      Decrease the account for the current sentiment and topic assignment of $w_i$
      $N_{d,l} \leftarrow N_{d,l} - 1, N_{l,k} \leftarrow N_{l,k} - 1, N_{d,l,k} \leftarrow N_{d,l,k} - 1, N_{l,k,i} \leftarrow N_{l,k,i} - 1, N_{k,i} \leftarrow N_{k,i} - 1$
      Sample a new sentiment label $l'$ and topic $k'$ for $w_i$ according to Eq. 16
      $N_{d,l'} \leftarrow N_{d,l'} + 1, N_{l',k'} \leftarrow N_{l',k'} + 1, N_{d,l',k'} \leftarrow N_{d,l',k'} + 1, N_{l',k',i} \leftarrow N_{l',k',i} + 1, N_{k',i} \leftarrow N_{k',i} + 1$
    end for
  end for
Run the Optimisation Step of Algorithm1 to update $\varphi$
end for
Update $\pi_{d,l}, \theta_{d,l,k}, \phi_{l,k,i}$ according to Eq. 17–19
```

3.4. Multi-label Topic Identification with Self-training

Self-training is a weakly-supervised algorithmic method for fine-tuning a base language model, improving the model's generalisation through generating pseudo-labels with high confidence scores and mitigates the coverage issue in weak supervision [42]. Compared to the semi-supervised models for text classification [43,44], which learn from small labelled documents, Shen et al. [45] developed the self-training algorithms with only label names. In this section, the fine-tuned topic and word embeddings and the conditional distributions of topics from the GJST could work perfectly as the base language model and the confidence score. Therefore, we propose a multi-label classifier with a self-training scheme based on the inference of the GJST to identify multiple topics mentioned in a document.

Through the Gibbs sampling procedure of the GJST model, we can obtain the embedding of topic k as $\mathbf{v}_k = x^C_{i=v_k}$ and the embedding of document d as $\mathbf{w}_d = \frac{1}{N_d}\sum_{w_j \in W_d} x^C_{i=w_j}$. We model the binary classifier for the topic k of document d as:

$$p_{k,d} = P(y_{k,d} = 1|d) = \sigma\left(\mathbf{v}_k \widetilde{W} \mathbf{w}_d^T\right) \tag{20}$$

where $\widetilde{W} \in \mathbb{R}^{h\times h}$ is a trainable matrix and $\sigma(\cdot)$ is a sigmoid function and $y_{k,d} = 1$ means that topic k is mentioned in document d. The intuition of the self-training is to iteratively select a high-confidence class or label for unlabelled data and utilise these pseudo-labelled data for training. Therefore, to train the classifier, we first generate pseudo-labels according to the confidence scores of samples. As presented in Section 3.3, the conditional distribution of topic k given a document d can be obtained as $\widetilde{P}(k|d) = \sum_l \theta_{d,l,k}$. Hence, the topic with a higher probability $\widetilde{P}(k|d)$ tends to be more confident.

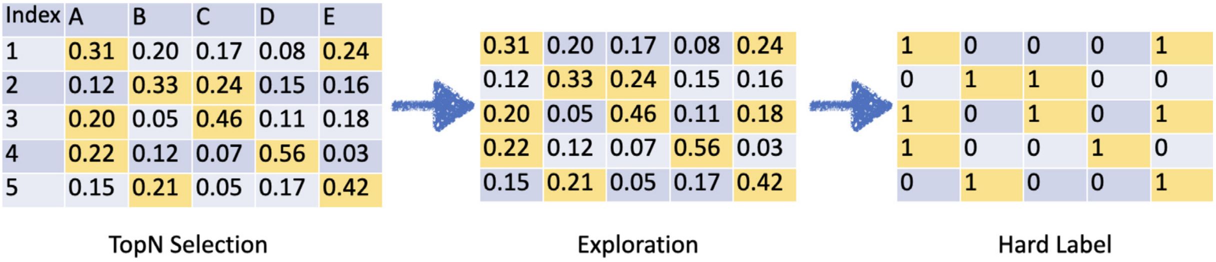
Pseudo-label Generation. As discussed earlier in the Introduction, selecting an appropriate threshold for the confidence score is a challenging task. Here, we introduce a straightforward method of generating pseudo labels, which adopts the $\widetilde{P}(k|d)$ as the confidence score for topics in a given document. We assign the *TopN* highest confident topics in each document with positive labels, i.e. $y_k = 1$ for $k \in \widetilde{K}_{TopN}$, where $\widetilde{K}_{TopN} = argmax_{TopN}\widetilde{P}(k|d)$ represents the set of *TopN* topics with the highest confidence. To select an appropriate *TopN*, we conduct a statistical analysis on $\widetilde{P}(k|d)$. We count the average number of confident topics with a probability greater than $1/K$, denoted by Avg_t, and set $TopN = Int(Avg_t)$where $Int(\cdot)$ means rounding to its nearest integer. As some less confident topics could also be relevant to the document, we sample positive labels from the set of less confident topics $\widetilde{K}_{\neg TopN}$ with an exploration probability ε. The procedure of generating pseudo positive labels Y_{pos} and pseudo negative labels Y_{neg} defined by *TopN* is presented in Algorithm3. A simple example illustrates how the pseudo-label generation strategy works, as shown in Fig. 3. In the example, each row represents a document with five topics. Assuming $TopN = 2$, we assign the two topics with the highest probabilities to positive labels for each document (yellow label). Then, an exploration probability, 0.2 in this case, is applied for each row. Row 3 is chosen for exploration. A topic is sampled from the candidate topics including $B3, D3$ and $E3$ according to their probabilities. As a result, topic $E3$ is selected and assigned with positive label.

Classifier Training. We train our classifier with the generated pseudo-labels by optimising the binary cross entropy loss as indicated in Eq. 21.

$$\mathscr{L}_{binary} = min - \sum_{d=1}^{D}\left(\sum_{k\in Y_{pos}} logp_{k,d} + \sum_{k\in Y_{neg}} log(1 - p_{k,d})\right) \tag{21}$$

Algorithm3: Pseudo-Label Generation

```
Require: Set of document embeddings $\{\mathbf{w}_d, d \in \mathscr{D}\}$, Set of topic embeddings $\{\mathbf{v}_k, k \in \{1,2,\dots K\}\}$, Set of conditional document-topic distributions $\left\{\widetilde{P}(k|d), k \in \{1,2,\dots K\}, d \in \mathscr{D}\right\}$, The number of topic selection $TopN$, The exploration probability $\varepsilon$
for document $d = 1$ to $D$ do
    Sort the set $\widetilde{K}_{TopN}$ and $\widetilde{K}_{\neg TopN}$
    for $k \in \widetilde{K}_{TopN}$ do
        if $random(\cdot) > \varepsilon$ then
            Assign topic $k$ as positive label, $y_{k,d} = 1$
            Update $Y_{pos,d} = Y_{pos,d} \cup \{k\}, \widetilde{K}_{TopN} = \widetilde{K}_{\neg TopN} \cap \{k\}$
        else
            Select $k'$ from $\widetilde{K}_{\neg TopN}$ according to $\widetilde{P}(k'|d)$
            Update $Y_{pos,d} = Y_{pos,d} \cup \{k'\}, \widetilde{K}_{\neg TopN} = \widetilde{K}_{\neg TopN} \cap \{k'\}$
        end if
        Assign all other topics with negative labels, $Y_{neg,d} = \{1,2,\dots,K\} \cap Y_{pos,d}, y_{k,d} = 0$ for $k \in Y_{neg,d}$
    end for
end for
```



Index	A	B	C	D	E
1	**0.31**	0.20	0.17	0.08	**0.24**
2	0.12	**0.33**	**0.24**	0.15	0.16
3	**0.20**	0.05	**0.46**	0.11	0.18
4	**0.22**	0.12	0.07	**0.56**	0.03
5	0.15	**0.21**	0.05	0.17	**0.42**

TopN Selection

0.31	0.20	0.17	0.08	**0.24**
0.12	**0.33**	**0.24**	0.15	0.16
0.20	0.05	**0.46**	0.11	**0.18**
0.22	0.12	0.07	**0.56**	0.03
0.15	**0.21**	0.05	0.17	**0.42**

Exploration

1	0	0	0	**1**
0	**1**	**1**	0	0
1	0	**1**	0	**1**
1	0	0	**1**	0
0	**1**	0	0	**1**

Hard Label

Fig. 3. Example of TopN Pseudo-label Generation.

The training with pseudo-labels only keeps the most confident labels for each sample, however, it also propagates the error of pseudo-labelling. To improve the generalisation of the classifier, we present a binary confidence regularisation to enhance high-confidence samples and diminish the errors of inaccurate predictions. The binary confidence regularisation iteratively computes a target distribution Q based on the model's current prediction P and aims to minimise the KL divergence between P and Q [46]:

$$\mathcal{L}_{self} = min\, KL(Q||P) = \sum_{d=1}^{D}\sum_{k=1}^{K} q_{k,d}\, log\frac{q_{k,d}}{p_{k,d}} \tag{22}$$

Following [45], $q_{k,d}$ is the soft pseudo-labels calculated by normalising the second power of $p_{k,d}$:

$$q_{k,d} = \frac{p_{k,d}^2 / \left(\sum_d p_{k,d}\right)}{p_{k,d}^2 / \left(\sum_d p_{k,d}\right) + (1 - p_{k,d})^2 / \left(\sum_d (1 - p_{k,d})\right)} \tag{23}$$

The intuition of Eq. 23 is to produce a higher target probability when the confidence level $p_{k,d}$ is high. We summarise the procedure of the self-training scheme in Algorithm 4, where *freq* denotes the frequency of updating the soft pseudo-labels.

Algorithm 4: Self-Training of the Multi-label Classifier

```
Require The initialized classifier $p_{k,d}$, the number of batches $B$
1: Generate pseudo-labels with Algorithm 3
2: Train the classifier $p_{k,d}$ with pseudo-labels by optimizing Eq. 21
3: for i from 1 to B do
4:    if i// freq = 0 then
5:       Update soft pseudo-labels $q_{k,d}$ with Eq. 23
6:    end if
7:    Train the classifier $p_{k,d}$ with $q_{k,d}$ by optimizing Eq. 22
8: end for
```

By obtaining the classifier of topic identification, we can further derive the probability of sentiment label l given that topic k^* is identified in document d^*:

$$p_{l,k,d} = P(l|y_{k^*} = 1, d^*) = \frac{\theta_{l,k=k^*,d=d^*}}{\sum_{l_i} \theta_{l_i,k=k^*,d=d^*}} \tag{24}$$

4. Experiment and Case Study

In this section, we aim to evaluate the performance of the proposed W-GJST model on both topic identification and topic-sentiment identification by comparing it to baseline methods. Two open-source datasets, namely *MAMS-ACSA* [47] and *Student Review Dataset* from Kaggle[1]. The experiment results will be analysed based on sentiment analysis and topic modelling metrics.

[1] https://www.kaggle.com/atk510/student-review-dataset

4.1. Datasets Description

The *MAMS-ACSA* is a multi-aspect multi-sentiment dataset designed for aspect category sentiment analysis. In the *MAMS-ACSA* dataset, eight topics are pre-defined, namely *food*, *service*, *staff*, *price*, *ambience*, *menu*, *place* and *miscellaneous*. Each sentence from *MAMS-ACSA* consists of at least two topics, and the average length of sentences is 9.8 words. Hence, the *MAMS-ACSA* dataset could evaluate the performance of our proposed model and the baseline methods on short texts. The sentiment polarity of each topic is labelled as "positive" and "negative". The *Student Review Dataset* is a collection of students' feedback on various perspectives of a University. Eight topics are identified in *Student Review Dataset (SRD)*: *job prospects*, *course and lecturers*, *students union*, *accommodation*, *uni facilities*, *local life*, *societies and sports* and *student support*. The original *Student Review Dataset* contains the texts from an overall review and reviews for each pre-defined topic. The *Student Review Dataset* is constructed by merging all the texts as one single review for each student.

Additionally, as the original sentiment polarity is labelled as a 5-point Likert scale, we categorise the sentiment polarity to labels of "positive" and "negative". As *MAMS-ACSA* is an imbalanced dataset, to evaluate the performance of models on the balanced dataset, we construct a sub-dataset from the *Student Review Dataset* by selecting samples with at least 4 topics mentioned and having balanced "positive" and "negative" labels. Preprocessing is conducted on both datasets consisting of the following steps: 1) remove all punctuations and numbers; 2) word lemmatisation; 3) remove the stop words. In our experiment, Natural Language Toolkit (NLTK)[2] is used for the preprocessing process in our experiment. Table 2 presents the statistical characteristics of datasets.

4.2. Baselines and Experiment Settings

To evaluate the performance of the W-GJST, we compare it with a variety of baseline models. Firstly, classic joint models including Sentiment-LDA [18], JST [6] and Reverse-JST (RJST) [48] are utilised. Weakly Supervised Topic Sentiment with Word Embedding (WS-TSWE) [11] is adopted to represent word embedding-based approaches. A semi-supervised Labelled Joint Sentiment Topic model (LJST) [21], which leverages a fraction of labelled training data, is introduced in addition to these weakly-supervised models. In the experiment, we examine the LJST with 20% and 50% of data labelled with overall document-level sentiments, denoted by LJST(20%) and LSJT(50%). To evaluate the effectiveness of the self-training topic classifier, we include an ablated version of W-GJST that identifies the topic through a pre-defined probability threshold for the conditional probability $\widetilde{P}(k|d)$ without adopting self-training scheme, denoted by W-GJST/ST.

For the proposed W-GJST model, we apply the Minka's fixed point iteration scheme [20] to update the parameter γ for each sentiment label every 5 iterations during the Gibbs sampling training process:

$$\gamma_l^t = \gamma_l^{t-1} \cdot \frac{\sum_d \Psi\left(N_{d,l} + \gamma_l^{t-1}\right) - D \cdot \Psi\left(\gamma_l^{t-1}\right)}{\sum_d \Psi\left(\sum_l N_{d,l} + \sum_l \gamma_l^{t-1}\right) - D \cdot \Psi\left(\sum_l \gamma_l^{t-1}\right)} \tag{25}$$

where γ_l^t represents the γ value for lth sentiment label at tth updating iteration and Ψ is the derivative of the gamma function, $\Psi(x) = \bigtriangledown_x \Gamma(x)$. We initialise $\gamma = 0.05\overline{N}/L$, where $\overline{N}$ is the average length of documents. According to many studies [11,21], the topic-sentiment-word prior is selected as $\beta = 0.01$. For document-sentiment-topic distribution, we set the parameter $\alpha = 50/K$. Additionally, to balance the sampling from Dirichlet multinomial distributions and the graph embedding-based multinomial distributions, the parameters for Bernoulli distributions are selected as $\lambda_1 = \lambda_2 = 0.1$. The learning rate of W-GJST is set as $\tau = 1e - 3$. The settings of parameters from their original studies are adopted for baseline models. We adopt the 300-dimensional Word2Vec word embedding vectors[3] trained on Wikipedia Corpus through the fastText Skipgram algorithm for W-GJST and WS-TSWE. In our experiment, we initialise the topic embeddings using the averaged word embeddings of the seed words of the pre-defined topics. Here, the seed words are defined as the words defining topics. For example, the initial embedding vector of topic *local life* in dataset SRD is formulated as the average embedding of words "local" and "life".

The experiment consists of three analyses to assess the models' performance on topic modelling, topic identification and topic-level sentiment classification. In the first set of analyses, we evaluate the performance of topic modelling under different numbers of Gibbs sampling iterations and different latent topic numbers. The number of latent topics for both datasets is set to $K \in \{5, 10, 20, 50\}$. The second analysis focuses on the classification performance on topic-sentiment identification of all models for two datasets considering their ground-truth labels. As the baseline models are initially designed for document-level sentiment analysis, we apply the same multi-label topic classifier and self-training scheme. The number of latent topics is set as $K = 8$ for both datasets. The settings of *TopN* are defined according to the $\widetilde{P}(k|d)$ obtained in the first analysis for each model. Another issue with classic joint models, i.e., Sentiment-LDA, JST and RJST, is how to associate the discovered latent topics with the pre-defined topics. We use cosine similarity in Eq. 1 to measure the similarity between each

[2] https://www.nltk.org

[3] http://vectors.nlpl.eu/repository

Table 2
Statistic Characteristics of Datasets.

Dataset	#Reviews	#Length	#Topics	#Vocab	#Pos	#Neg	#None
MAMS-ACSA	3,149	9.70	8	5,248	1,929	5,161	18,102
SRD	5,598	55.53	8	13,505	13,525	11,591	19,668

latent topic's top N_{top} related words and the words describing pre-defined topics. Each pre-defined topic is represented by its most similar latent topic. In the third analysis, we examine the impacts of the parameter λ and the learning rate of ConvNet τ on the proposed W-GJST model in terms of both topic modelling and identification and topic-sentiment classification. For our experiment, each testing model runs 50 iterations of Gibbs sampling, and the results are reported based on the average over 5 rounds. Additionally,

4.3. Evaluation Metrics

In our experiment, we aim to evaluate the performance of all testing models on the multi-topic sentiment analysis. The evaluation assesses the quality of topic modelling, the accuracy of topic identification and the topic-sentiment identification. Two evaluation metrics are introduced to investigate the quality of topics discovered by all testing models: topic coherence score (TCS) and H-score. Proposed by [49], TCS is a word co-occurrence based measurement for topic quality. Given a topic k and its top N_{top} related words $\mathscr{V}^k = \left(w_1^k, \ldots, w_{N_{top}}^k\right)$ ordered by $p(w|z=k)$, the average TCS is defined as:

$$TCS = \frac{1}{K}\sum_{k=1}^{K} TCS\left(z=k; \mathscr{V}^k\right) = \sum_{t=2}^{T}\sum_{i=1}^{t} log\frac{D(w_t^k, w_i^k)+1}{D(w_i^k)} \tag{26}$$

where $D(w_t^k, w_i^k)$ is the number of documents with the co-occurrence of word w_t^k and w_i^k and $D(w_i^k)$ is the number of documents containing the word w_i^k. TCS is based on the idea that words tend to co-occur in the same document if they belong to a single topic and the larger TCS value indicates that topics are more coherent [21]. Another topic quality metric is H-score based on the Jensen-Leibler divergence [49]. The H-score calculates the ratio of the intra-cluster distance $IntraDis(C)$ to the inter-cluster distance $InterDis(C)$ of document clusters:

$$H - score = \frac{IntraDis(C)}{InterDis(C)} \tag{27}$$

$$IntraDis(C) = \frac{1}{K}\sum_{k=1}^{K}\left[\sum_{d_i, d_j \in C_k}\frac{2dist(d_i, d_j)}{|C_k| \cdot |C_k - 1|}\right] \tag{28}$$

$$InterDis(C) = \frac{1}{K(K-1)}\sum_{C_k, C_{k'} \in C}\left[\sum_{d_i \in C_k}\sum_{d_j \in C_{k'}}\frac{2dist(d_i, d_j)}{|C_k| \cdot |C_{k'}|}\right] \tag{29}$$

where $C = \{C_1, \ldots, C_K\}$ is the set of document clusters based on the topics, d_i is the vectorized representation of ith document formed by the posterior distribution of topics $d_i = [p(z_1|d_i), \ldots, p(z_K|d_i)]$, and $dist(d_i, d_j)$ represents the Jensen-Leibler divergence of d_i and d_j. A lower H-score indicates that the average inter-cluster distance is larger than the intra-cluster distance, which implies that the clusters are tightly coupled and desired [21].

To evaluate the quality of topic identification and topic-sentiment identification, precision, recall and F1-score are used as evaluation metrics, which can be calculated as:

$$Precision = \frac{TP}{TP + FP} \tag{30}$$

$$Recall = \frac{TP}{TP + FN} \tag{31}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{32}$$

where TP refers to the number of true positive, i.e. the number of positive data correctly classified as positive, FP refers to the number of false positive, i.e., the number of negative data falsely classified as positive, and FN refers to false negative, i.e., the number of positive data falsely labelled as negative. A larger value of Precision, Recall and F1 indicate a better performance and these values are computed using a macro-average.

4.4. Experiment Results

This section presents and discusses the experimental results of three analyses, namely the quality of topic modelling, document-level topic identification and sentiment-topic identification, and the impacts of hyper-parameters λ and τ.

4.4.1. Evaluation on Topic Modelling

This subsection evaluates the performance on the quality of topic modelling for all methods. Table 3 and Table 4 present the topic coherence score (TCS) results for sentiment-LDA, JST, RJST, WS-TSWE, LJST and the proposed W-GJST models with a different number of latent topics. The numbers of most related words are set as 10 and 20 respectively for Table 3 and 4. As we can observe, both Tables demonstrate that our proposed W-GJST model significantly outperforms baseline models on all datasets. Specifically, in Table 3, the proposed W-GJST model achieves 1.1587 on TCS with 5 latent topics, 8.3% and 5.5% higher than JST and WS-TSWE on the MAMS-ACSA dataset, respectively. This figure of the W-GJST is improved to 2.8528 and 4.6313 for $K = 20$ and $K = 50$. In contrast, as the best two performers among the baseline models, WS-TSWE and LJST achieve similar results on TCS for the MAMS-ACSA dataset. WS-TSWE achieves a 18.1% lower value for $K = 20$ and a 19.1% lower value for $K = 50$. The better performance of our proposed W-GJST model is further demonstrated on the SRD dataset in Table 3. The TCS values for RJST and WS-TSWE for $K = 10$ are 0.5996 and 0.6883, respectively, whereas the W-GJST model gets a value of 1.6931. For $K = 50$, the TCS value for W-GJST model is 4.1796, which is 46.6% and 41.98% higher than that of WS-TSWE and LJST, respectively. Similar results can be found in Table 4 with $N_{top} = 20$. The proposed W-GJST obtains the highest TCS values of 5.8818 and 5.8150 for MAMS-ACSA and SRD datasets with $K = 50$. The latter value is 39.2% higher than the TCS value of WS-TSWE on the SRD dataset. Fig. 4 and Fig. 5 visualise the TCS values for all models on MAMC-ACSA and SRD, respectively. It can be observed from Table 3 and Table 4, the TCS values of all models improve with the increment of the number of topics (K). Additionally, the TCS values for $N_{top} = 20$ is higher than the values for $N_{top} = 10$. For instance, RJST model achieves a TCS value of 2.0961 for $K = 10$ and $N_{top} = 20$ on MAMS-ACSA dataset, which is 26.9% higher than the corresponding value for $N_{top} = 10$. Figs. 4 and 5 illustrate the comparison of TCS scores for all models.

Table 5 and Fig. 6 present the H-score performance of all models on both datasets with different numbers of topics. As we can observe from Table 5, the proposed W-GJST model achieves the lowest values of H-score on all cases for both datasets. For the MAMS-ACSA dataset, the best H-score obtained by the W-GJST model is 0.1267 at $K = 5$, which is slightly lower than the value of 0.1297 for WS-TSWE, followed by the JST model. Sentiment-LDA obtains an H-score of 0.4578 at $K = 50$, more than three times higher than the W-GJST model with a value of 0.1410. For the SRD dataset, WS-TSWE has the best performance on the H-score among all baseline models, with the lowest H-score of 0.1533 at $K = 10$. Comparatively, W-GJST achieves a lower value of 0.1325 at $K = 10$. RJST and Sentiment-LDA are the worst performed models, and the RJST model obtains the highest H-score of 0.5886 at $K = 50$ which is four times higher than the value of the W-GJST. As indicated by Fig. 6, H-score increases with the increment of K for both datasets, except for the proposed W-GJST model on SRD dataset. However, the variation of H-score values obtained by the W-GJST model is small. Next, we compare the computation time per iteration of Gibbs Sampling for each model, as shown in Table 6. The LJST model has the shortest computation time per iteration for MAMS-ACSA, and the JST model is the fastest model for the SRD dataset. The WS-TSWE is the most time-

Table 3
Topic Coherence Score on Two Datasets with Different Latent Topics ($N_{top} = 10$).

Model	**MAMS-ACSA**				**SRD**			
	$K = 5$	$K = 10$	$K = 20$	$K = 50$	$K = 5$	$K = 10$	$K = 20$	$K = 50$
Sentiment-LDA	1.0301	1.3769	1.8337	3.0755	0.4133	0.4693	0.7735	1.3694
JST	1.0621	1.5341	2.1813	3.6956	0.5099	0.5662	0.8057	1.4138
RJST	1.0414	1.5304	2.1464	3.6739	0.4363	0.5996	0.8232	1.4097
WS-TSWE	1.0948	1.5267	2.3364	3.7485	0.5527	0.6883	1.3587	2.2335
LJST	1.0693	1.6345	2.3791	3.6857	0.5294	0.6415	1.3906	2.1252
W-GJST	**1.1587**	**1.9243**	**2.8528**	**4.6313**	**0.6499**	**1.6931**	**2.5739**	**4.1796**

Table 4
Topic Coherence Score on Two Datasets with Different Latent Topics ($N_{top} = 20$).

Model	**MAMS-ACSA**				**SRD**			
	$K = 5$	$K = 10$	$K = 20$	$K = 50$	$K = 5$	$K = 10$	$K = 20$	$K = 50$
Sentiment-LDA	1.0312	1.7783	3.0692	5.0636	0.4233	0.7118	0.9777	2.3079
JST	1.3708	2.0481	3.1119	5.0975	0.6158	0.7010	1.1110	2.4259
RJST	1.3545	2.0961	3.0692	5.0636	0.4784	0.7776	1.1925	2.3818
WS-TSWE	1.3319	2.2132	3.2506	5.1539	0.7932	0.9455	2.1135	3.5379
LJST	1.4138	2.5197	3.5864	5.3233	0.7624	0.8859	2.2133	3.2327
W-GJST	**1.7454**	**2.7566**	**3.7701**	**5.8818**	**0.9497**	**1.6568**	**3.5304**	**5.8150**

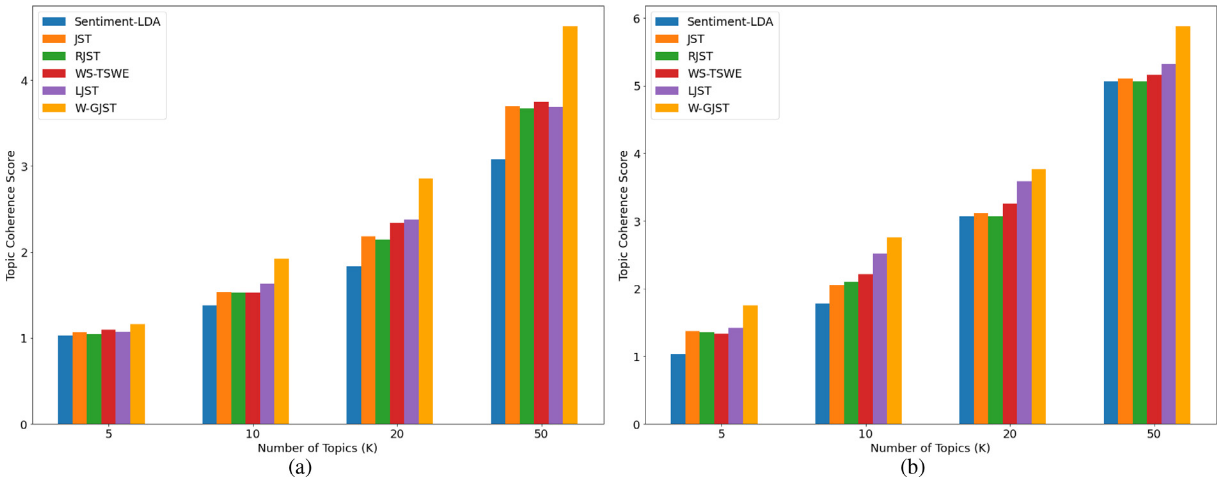


Fig. 4. Topic Coherence Scores on MAMC-ACSA with (a) top 10 related words of each topic (i.e. $N_{top} = 10$) (b) top 20 related words of each topic (i.e. $N_{top} = 20$).

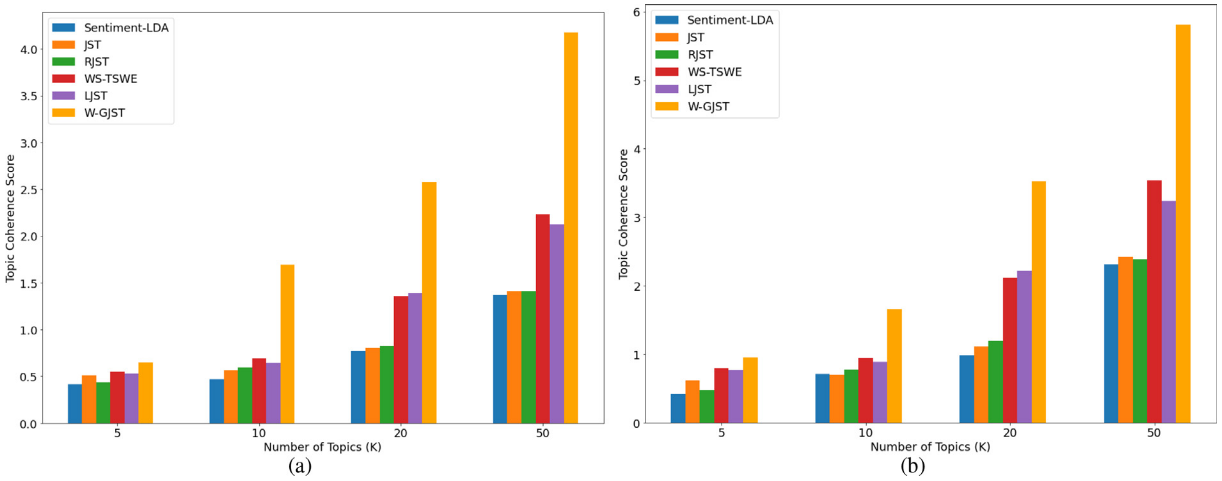


Fig. 5. Topic Coherence Scores on SRD with (a) top 10 related words of each topic (i.e. $N_{top} = 10$) (b) top 20 related words of each topic (i.e. $N_{top} = 20$).

Table 5
H-Score on Two Datasets with Different Latent Topics.

Model	**MAMS-ACSA**				**SRD**			
	$K = 5$	$K = 10$	$K = 20$	$K = 50$	$K = 5$	$K = 10$	$K = 20$	$K = 50$
Sentiment-LDA	0.2399	0.3473	0.3667	0.4578	0.3539	0.4609	0.5228	0.5744
JST	0.1461	0.1491	0.1653	0.1787	0.1864	0.2023	0.2199	0.2314
RJST	0.2495	0.3038	0.3578	0.4237	0.3367	0.4373	0.4988	0.5886
WS-TSWE	0.1297	0.1321	0.1356	0.1529	0.1584	0.1533	0.1770	0.2115
LJST	0.2032	0.2177	0.2272	0.2341	0.2045	0.2221	0.2387	0.2453
W-GJST	**0.1267**	**0.1246**	**0.1254**	**0.1410**	**0.1553**	**0.1325**	**0.1347**	**0.1323**

consuming model for both datasets. With the implementation of importance sampling for the W-GJST model, we can observe that it performs much faster than WS-TSWE, especially for the large dataset, i.e., SRD.

Combining the results of both TCS and H-score, we observe that the word embedding enhanced models, i.e., W-GJST and WS-TSWE, tend to have a better performance on topic modelling. The WS-TSWE has higher TCS values compared to classic joint models, such as JST and RJST. Even though the LJST has a competitive performance on TCS, it has much higher H-score values than W-GJST and WS-TSWE. It indicates that the utilisation of training data labelled with sentiment labels does not significantly improve the models' ability to discover the topic-document relationship. The proposed W-GJST shows the best

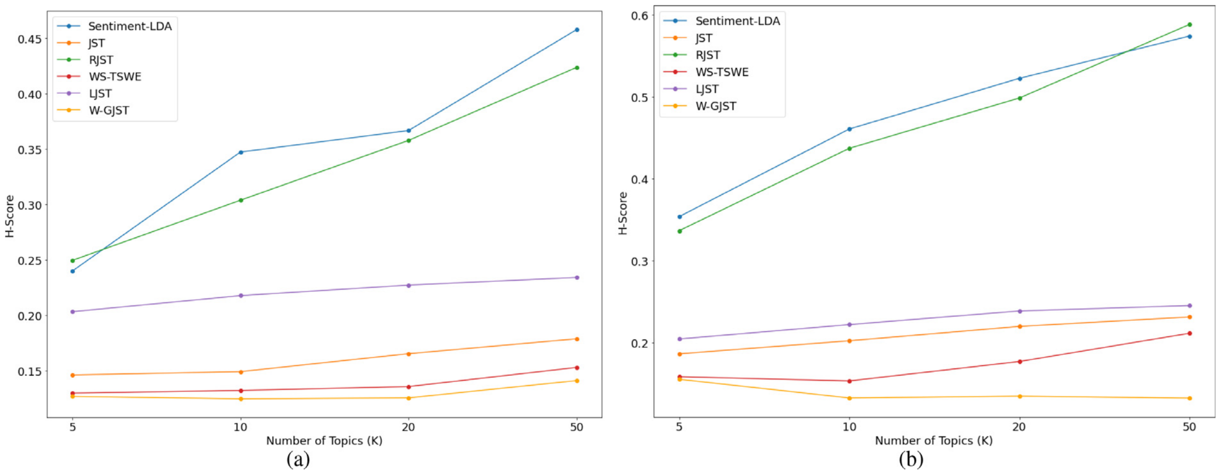


Fig. 6. H-Score on Different Topics (a) MAMS-ACSA (b) SRD.

Table 6
Computation Time (Gibbs Sampling) of Topic Model-based Methods.

Dataset	Computation Time (seconds/ iteration)					
	Sentiment-LDA	JST	RJST	WS-TSWE	LJST	W-GJST
MAMS-ACSA	4.5991	2.8420	4.2971	21.5202	2.4052	3.3993
SRD	31.6595	21.2298	32.1874	268.0750	25.3586	26.9171

performance on both TCS and H-score. Compared to its counterpart WS-TSWE, the fine-tuned word and topic embeddings trained through the E-GCN could provide more accurate word-to-word and word-to-topic relations, supported by its results with higher TCS and lower H-score. Additionally, the importance sampling-based training method enables the W-GJST to have an efficient computational ability. To summarise, the proposed W-GJST model outperforms all baseline models on the quality of topic modelling.

4.4.2. Evaluation on Topic and Sentiment Classification

In the second set of the experiment, we compare the performance on the topic identification and topic-sentiment identification of our proposed W-GJST model with baseline models, shown in Table 7. As the baseline models can only perform document-level sentiment analysis, we deploy the same multi-topic classifier and self-training scheme to them for a fair examination of how the learned joint models could contribute to the task of topic and sentiment identification. After the Gibbs Sampling training of these models, we feed the obtained joint sentiment-topic distribution given a document, i.e. $\theta_{d,l,k}$, into the multi-topic classifier. The trained word and topic embeddings are applied to build the probability of a topic given a document $p_{k,d}$ in Eq. 20. By analysing the obtained conditional probability of topics $\widetilde{P}(k|d)$ for each model, we define the values of *TopN* as 2 and 3 for datasets MAMS-ACSA and SRD, respectively. Fig. 7 shows an example of the distributions of the number of confident topics and the deviation of $\widetilde{P}(k|d)$ for MAMS-ACSA. For these presented models, the majority of the number of confident topics fall into the range of $[2,3]$. By rounding the average number of confident topics Avg_t, we can set $TopN = 2$ for MAMS-ACSA. In our experiment, we set the same value of *TopN* for all models as they have similar distributions of the number of confident topics. However, *TopN* can be different for each model as their $\widetilde{P}(k|d)$ could be different. The training procedure of $p_{k,d}$ follows the description presented in Section 3.4. Furthermore, we examine the ablated W-GJST/ST to evaluate the effectiveness of the proposed multi-topic classifier with a self-training scheme. We apply a pre-defined probability threshold of 0.2 on the $\widetilde{P}(k|d)$ obtained through the GJST module, and assign $y_{k,d} = 1$ if $\widetilde{P}(k|d) > 0.2$. The setting of probability threshold is defined based on the deviation of $\widetilde{P}(k|d)$, shown in Fig. 7. Once the task of topic identification is completed, Eq. 24 can be applied to identify the corresponding sentiments.

In Table 7, the RJST model achieves better results in all metrics for identifying topics mentioned in the documents than the Sentiment-LDA model and JST model on both datasets. More specifically, for the MAMS-ACSA dataset, the accuracies and F1 scores of topic identification obtained by the RJST model are 58.33% and 57.75%, which are 3.28% and 4.98% higher than the values of the JST model, and 12.87% and 12.42% higher than that of Sentiment-LDA. Because of the integration with pre-trained word embeddings, the WS-TSWE outperforms RJST with 11.89% higher F1 score for topic identification of the MAMS-ACSA dataset. By increasing the amount of labelled data used in LJST, the performance on topic identification of the LJST (50%) is improved accordingly. The F1 score of LJST (50%) on the MAMS-ACSA dataset, i.e. 0.6072, is slightly lower than that

Table 7
Classification Results on Two Datasets

Dataset	**Model**	Topic Identification				Topic-Sentiment Identification			
		Acc	Precision	Recall	F1	Acc	Precision	Recall	F1
MAMS-ACSA	Sentiment-LDA	0.4546	0.5113	0.5104	0.4533	0.3812	0.3476	0.3552	0.3514
	JST	0.5505	0.5464	0.5558	0.5277	0.5029	0.3882	0.3803	0.3842
	RJST	0.5833	0.5703	0.5850	0.5775	0.5139	0.3633	0.3704	0.3668
	WS-TSWE	0.6293	0.5937	0.6043	0.5989	0.5737	0.3933	0.4033	0.3983
	LJST (20%)	0.5863	0.5837	0.5716	0.5773	0.5607	0.3853	0.3979	0.3914
	LJST (50%)	0.6163	0.6045	0.6112	0.6072	0.5611	0.4023	0.4142	0.4081
	W-GJST	**0.7049**	**0.6250**	**0.6227**	**0.6238**	**0.6557**	**0.4267**	**0.4205**	**0.4236**
	W-GJST/ST	0.6196	0.5063	0.4927	0.4994	0.5931	0.3345	0.3555	0.3447
SRD	Sentiment-LDA	0.6036	0.6253	0.5647	0.5155	0.3339	0.4167	0.3733	0.3188
	JST	0.6089	0.6047	0.6008	0.6027	0.4077	0.4145	0.4033	0.3827
	RJST	0.6109	0.6138	0.5905	0.6019	0.3756	0.4133	0.3963	0.3674
	WS-TSWE	0.6137	0.6062	0.6043	0.6052	0.4209	0.4135	0.4103	0.4119
	LJST (20%)	0.5972	0.5903	0.5752	0.5826	0.3921	0.3977	0.3861	0.3918
	LJST (50%)	0.6092	0.5939	0.5875	0.5907	0.4301	0.4033	0.4024	0.4028
	W-GJST	**0.6270**	**0.6115**	**0.6133**	**0.6124**	**0.4437**	**0.4255**	**0.4276**	**0.4265**
	W-GJST/ST	0.5761	0.5148	0.5176	0.5162	0.3935	0.4145	0.3709	0.3514

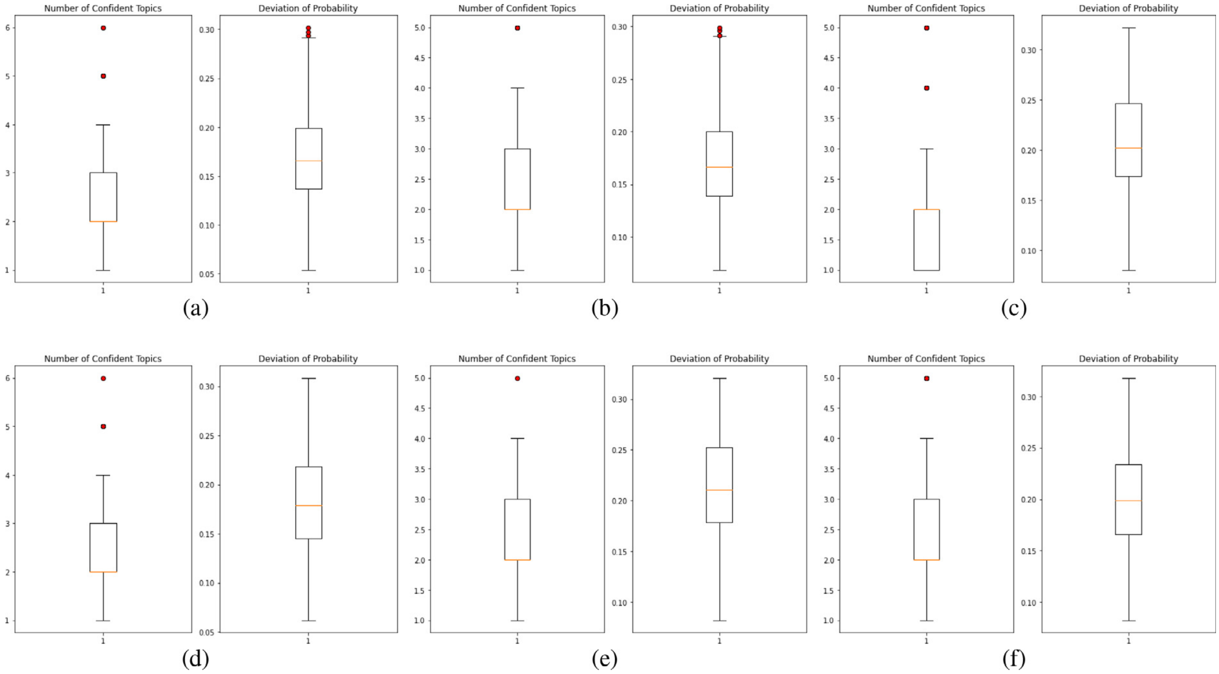


Fig. 7. The Statistic Characteristics of $\widetilde{P}(k|d)$, i.e. the Number of Confident Topics and the Deviation of Probability, for (a) W-GJST (b) JST (c) RJST (d) WS-TSWE (e) LJST (f) Sentiment-LDA.

of the WS-TSWE. The W-GJST model achieves the highest accuracy of topic identification (i.e. 70.49%) among all testing models, compared to the value of 62.93% obtained by the WS-TSWE model. The experiment of topic identification on the SRD dataset reveals similar results. WS-TSWE model has the best performance among all baseline models with an accuracy of 61.37% and an F1 score of 60.52%. However, the proposed W-GJST model outperforms WS-TSWE on the accuracy, precision, recall and F1 score of 62.70%, 61.15%, 61.33% and 61.24%.

The task of topic-sentiment identification can be regarded as a subsequent task of topic identification according to Section 3.4. Therefore, it evaluates the quality of the parameter inference, i.e. the estimation of $\theta_{d,l,k}$, and the accuracy of topic identification. As MAMS-ACSA is an unbalanced dataset, we can observe that the F1 scores of models are significantly lower than their corresponding accuracies. In this case, the F1 score is a more reliable metric. Sentiment-LDA performs the worst on topic-sentiment identification with an F1 score of 35.14%. LJST (50%) performs the best among baseline models with an F1 score of 40.81%, followed by the WS-TSWE, JST and RJST models. Nevertheless, the proposed W-GJST model outmatches the baseline models on both accuracy and F1 score. According to Table 2, the distributions of both topics and sentiment labels in the SRD dataset are more balanced. In this case, WS-TSWE and W-GJST get accuracies and F1 scores over 40%, while the F1 score of W-GJST is 1.49% higher than that of WS-TSWE. In addition, the performance of W-GJST significantly deteriorates when removing the multi-topic classifier with self-training. The F1 scores of the W-GJST/ST on the MAMS-ACSA dataset decrease by 19.94% for the topic identification and 18.63% for the topic-sentiment identification.

LJST achieves a similar performance with the WS-TSWE on both topic identification and sentiment identification. We can observe that the contribution of label information to the sentiment identification is not as significant as expected. The LJST only leverages the document-level sentiment labels, which is not sufficiently informative when dealing with the fine-grained multi-topic sentiment analysis. Therefore, we can conclude that the integration of pre-trained word embeddings may assist the identification of topics due to the better performance of WS-TSWE and W-GJST comparing to other baseline models. The design and implementation of a graph neural network in the W-GJST model could exploit the relations between topics and documents. Moreover, the effectiveness of the proposed multi-topic classifier with self-training scheme can be demonstrated by the improvement of performance on multi-label topic identification and topic-sentiment identification.

To further evaluate and visualise the results on topic and sentiment identification, Tabel 8 and Table 9 indicate the 5 most related words per sentiment label for each pre-defined topic of the SRD dataset for the WS-TSWE model and W-GJST model, respectively. "P" and "N" represent the positive sentiment label and the negative sentiment label of each topic. All the words are obtained based on the distribution $\phi_{l,k,i}$ presented in Eq. 19. As we can observe, our proposed W-GJST model could recognise the topic words and sentiment words more accurately than WS-TSWE. For example, "interest", "living" and "life" are defined as the positive topic words for *Job Prospects* by WS-TSWE. In contrast, more related words such as "work", "employability" and "interview" are identified by the W-GJST model. For the topic of *Uni Facilities*, we can observe that words such as "parking", "centre", "shop" and "café" are recognised by the W-GJST model. These words describe the *Uni Facilities* more accurately compared to "film", "studio", "book" and "idea" identified by the WS-TSWE. In addition, both models can extract sentiment words simultaneously, as shown in Tables 8 and 9. However, the proposed W-GJST model tends to identify more suitable sentiment words for each topic. For instance, "supportive" and "boring" for *Student Union*, "small" for *Accommoda-*

Table 8
Extracted words related to each topic under different sentiment labels by WS-TSWE on SRD.

Topic	Sentiment	Most Related Words				
Job Prospects	P	interest	living	**amaze**	life	load
	N	work	placement	module	lecture	time
Course and Lectures	P	tutor	industry	class	lesson	feedback
	N	teach	**poor**	course	terrible	project
Student Union	P	student	system	organize	**supportive**	environment
	N	workshop	employability	session	law	information
Accommodation	P	accommodation	campus	location	town	bus
	N	facility	food	parking	computer	park
Uni Facilities	P	move	average	**hate**	film	school
	N	studio	equipment	book	idea	side
Local Life	P	kitchen	mark	pay	day	left
	N	room	hall	flat	bathroom	door
Societies and Sports	P	health	service	society	**pretty**	**helpful**
	N	club	service	uni	people	career
Student Support	P	support	experience	practical	knowledge	**excellent**
	N	offer	opportunity	academic	activity	**lack**

Table 9
Extracted words related to each topic under different sentiment labels by W-GJST on SRD.

Topic	Sentiment	Most Related Words				
Job Prospects	P	work	employability	workshop	seminar	interview
	N	volunteer	cv	question	session	apply
Course and Lectures	P	coursework	resource	department	technology	extracurricular
	N	theory	practice	personal	department	**heavy**
Student Union	P	**supportive**	knowledge	learner	practical	**excellent**
	N	powerpoint	gender	**boring**	practical	**harder**
Accommodation	P	accommodation	campus	location	bus	uni
	N	room	place	**small**	disability	space
Uni Facilities	P	facility	travel	parking	centre	shop
	N	health	cafe	contact	timetable	commute
Local Life	P	club	library	event	**enjoyable**	pub
	N	career	variety	accessibility	nightlife	security
Societies and Sports	P	sport	hockey	**friendly**	capacity	organize
	N	game	dance	doctor	issue	welfare
Student Support	P	tutor	opportunity	**support**	industry	skill
	N	feedback	lesson	profession	staff	**lack**

tion, “enjoyable” for *Local Life* and “friendly” for *Societies and Sports*. In conclusion, the above results and analysis demonstrate the effectiveness of the W-GJST model in discovering the relations between words and topics.

4.4.3. Impacts of Key Parameters

In this section, we investigate the impacts of three important parameters of the proposed W-GJST model, namely the Bernoulli distribution parameter λ, the learning rate of graph representation τ and the value of *TopN*, on the performance of topic modelling, topic identification and topic-sentiment identification for the MAMS-ACSA dataset. Fig. 8 shows the variation of metrics with different values of λ, and the learning rate τ is set as $2e-3$. As we can observe, the accuracies of both topic identification and topic-sentiment identification increase with the increment of λ from 0.1 to 0.5 and reach the highest values when $\lambda = 0.5$. After that, accuracies decrease with λ varies from 0.5 to 0.9. The topic coherence score rises steadily from 2.3897 to 5.7756, with λ increases from 0.1 to 0.7. Meanwhile, the variation of the H-score is comparatively small. The H-score at $\lambda = 0.4$ reaches the lowest value of 0.1070, and the highest H-score (0.1432) is obtained at $\lambda = 1$. By fixing $\lambda = 0.1$, we run the experiment of the W-GJST model on the MAMS-ACSA dataset with a learning rate τ varying between $[5e-4, 6e-3]$, as shown in Fig. 9. The accuracy of topic identification and topic-sentiment identification fluctuates with the increase of τ, and both reach the best outcomes at $\tau = 1e-3$. Comparatively, the variations of topic coherence score and H-score are not as significant as accuracies. The difference between the highest and the lowest H-score is only 0.01. The results indicate that the W-GJST could achieve better results on all metrics when $\lambda = 0.5$ with a fixed learning rate. Comparatively, the value of the learning rate τ has a stronger influence on the accuracy of topic and sentiment identification. Due to the weakly supervision of the proposed W-GJST, fine-tuning these two parameters for an unknown dataset could be challenging. It is suggested to set $\lambda = 0.5$ to balance the sources of the word sampling during the training procedure of GJST. As for the learning rate, it is suggested to start with $\tau = 1e-3$ according to our experiment results. To further fine-tune the τ,

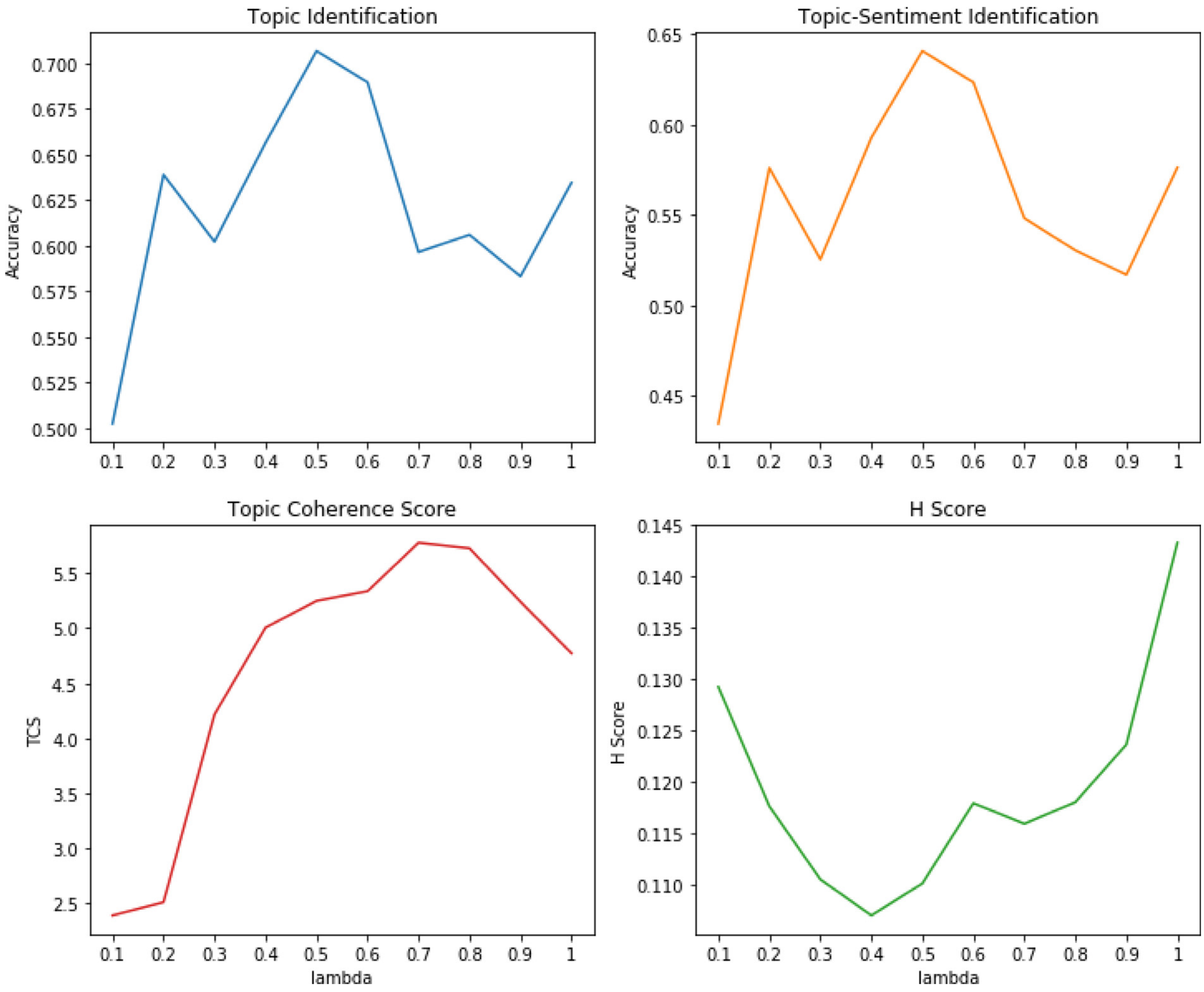


Fig. 8. Evaluation metrics under different λ values (learning rate: $\tau = 2e-3$).

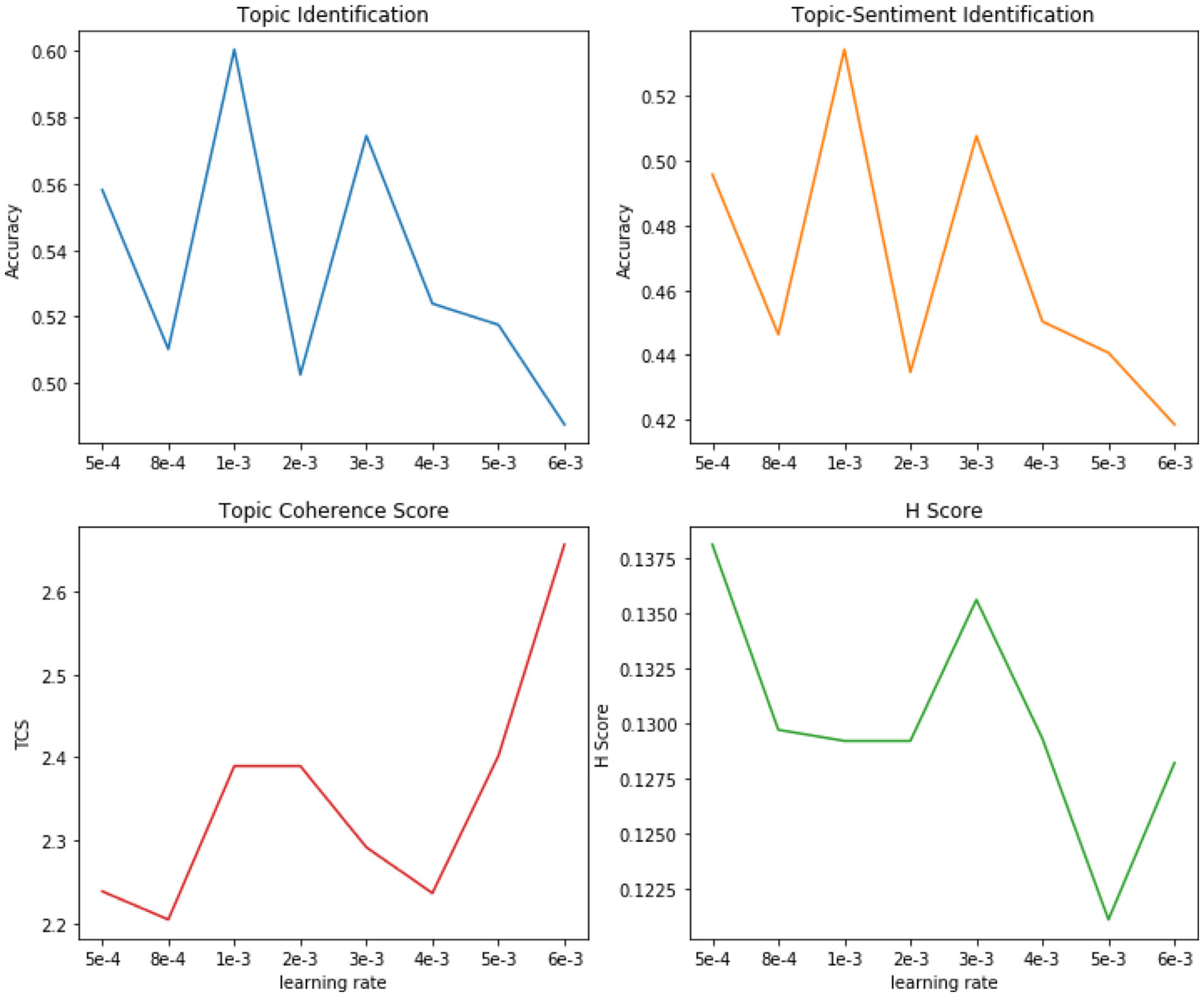


Fig. 9. Evaluation metrics under different learning rates τ ($\lambda = 0.1$).

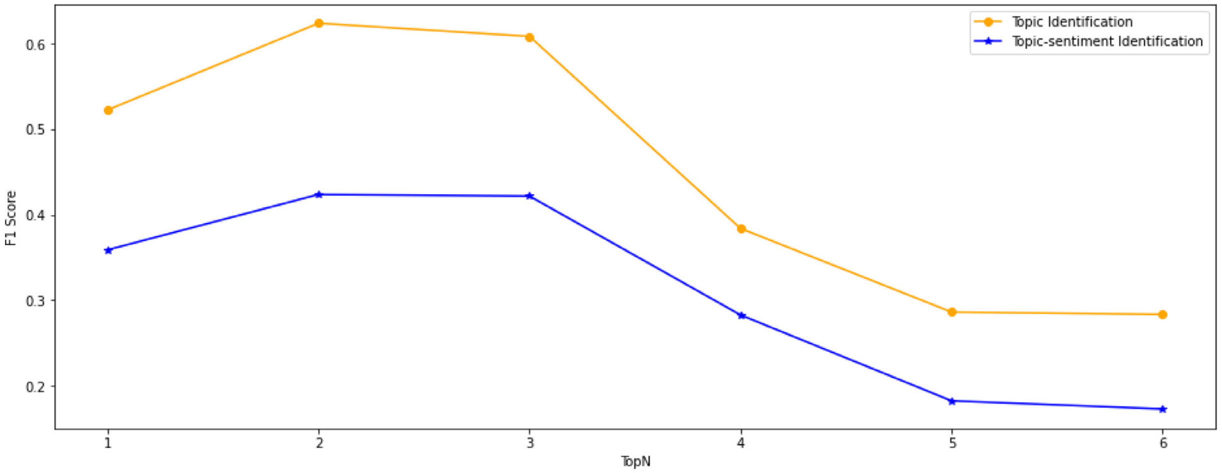


Fig. 10. F1 Score under different values of TopN.

we can either manually annotate a small amount of testing data for evaluation or monitor the change of the objective function 22. Apart from λ and τ, we also analyse how the selection of *TopN* could impact the performance of W-GJST. Fig. 10 shows the variation of F1 scores for topic identification and topic-sentiment identification with different values of *TopN*. It is obvious that the W-GJST performs the best when $TopN = 2$ and $TopN = 3$, which demonstrates our method of selecting the appropriate value of *TopN* shown in Fig. 7. A *TopN* value smaller or greater than the interquartile range could result in the imbalance pseudo labels that hinders the performance of our model.

5. Conclusions and Future Work

This paper proposes a novel model for the multi-topic sentiment analysis named the weakly-supervised graph-based joint sentiment topic model (W-GJST). The W-GJST consists of a graph-based joint sentiment-topic model (GJST) and a multi-topic classifier. The GJST incorporates the graph representations of topics and words and a joint sentiment-topic model by introducing an extra embedding component. We propose an Edge-gated Graph Convolutional Network (E-GCN) to learn the distributed topic embeddings. An importance sampling-based training scheme for the E-GCN is adopted to reduce the computational complexity of updating large graph representations. To perform the multi-topic identification, we first train the GJST with the Gibbs sampling algorithm, and build a multi-label classifier based on the learned topic and word embeddings. A self-training method is proposed to train the classifier with unlabelled data.

The superiority of the proposed W-GJST is validated by comparing to baseline models from perspectives of topic modelling, topic and sentiment classification. The results demonstrate that the W-GJST has better modelling on topic and word distributions and faster processing time and, can perform on the fine-grained topic-sentiment classification with limited supervision and satisfactory results. The results also indicate the practical value and business significance of implementing the W-GJST in real-word applications, such as automated text mining and classification. In the future, we will consider researching the self-training method without the pre-defined threshold or with the adjustable threshold for generating pseudo-labels.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] L. Gong, L. Lin, W. Song, H. Wang, Jnet: Learning user representations via joint network embedding and topic embedding, in: Proceedings of the 13th International Conference on Web Search and Data Mining, 2020, pp. 205–213.
[2] D. Hyun, C. Park, M.-C. Yang, I. Song, J.-T. Lee, H. Yu, Target-aware convolutional neural network for target-level sentiment analysis, Information Sciences 491 (2019) 166–178.
[3] Y. Wang, M. Huang, X. Zhu, L. Zhao, Attention-based lstm for aspect-level sentiment classification, in: Proceedings of the 2016 conference on empirical methods in natural language processing, 2016, pp. 606–615.
[4] C. Wan, Y. Peng, K. Xiao, X. Liu, T. Jiang, D. Liu, An association-constrained lda model for joint extraction of product aspects and opinions, Information Sciences 519 (2020) 243–259.
[5] B. Ozyurt, M.A. Akcayol, A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: Ss-lda, Expert Systems with Applications 168 (2021) 114231.
[6] C. Lin, Y. He, Joint sentiment/topic model for sentiment analysis, in: Proceedings of the 18th ACM conference on Information and knowledge management, 2009, pp. 375–384.
[7] L.-Y. Dong, S.-J. Ji, C.-J. Zhang, Q. Zhang, D.W. Chiu, L.-Q. Qiu, D. Li, An unsupervised topic-sentiment joint probabilistic model for detecting deceptive reviews, Expert Systems with Applications 114 (2018) 210–223.
[8] F. Tang, L. Fu, B. Yao, W. Xu, Aspect based fine-grained sentiment analysis for online reviews, Information Sciences 488 (2019) 190–204.
[9] D.Q. Nguyen, R. Billingsley, L. Du, M. Johnson, Improving topic models with latent feature word representations, Transactions of the Association for, Computational Linguistics 3 (2015) 299–313.
[10] A. García-Pablos, M. Cuadros, G. Rigau, W2vlda: almost unsupervised system for aspect based sentiment analysis, Expert Systems with Applications 91 (2018) 127–137.
[11] X. Fu, X. Sun, H. Wu, L. Cui, J.Z. Huang, Weakly supervised topic sentiment joint model with word embeddings, Knowledge-Based Systems 147 (2018) 43–54.
[12] D. Shen, C. Qin, C. Wang, Z. Dong, H. Zhu, H. Xiong, Topic modeling revisited: A document graph-based neural network perspective, Advances in Neural Information Processing Systems 34 (2021).
[13] F. Zhuang, G. Karypis, X. Ning, Q. He, Z. Shi, Multi-view learning via probabilistic latent semantic analysis, Information Sciences 199 (2012) 20–30.
[14] Q. Mei, X. Ling, M. Wondra, H. Su, C. Zhai, Topic sentiment mixture: modeling facets and opinions in weblogs, in: in: Proceedings of the 16th international conference on World Wide Web, 2007, pp. 171–180.
[15] C. Lin, Y. He, R. Everson, S. Ruger, Weakly supervised joint sentiment-topic detection from text, IEEE Transactions on Knowledge and Data engineering 24 (6) (2011) 1134–1145.
[16] Y. Jo, A.H. Oh, Aspect and sentiment unification model for online review analysis, in: Proceedings of the fourth ACM international conference on Web search and data mining, 2011, pp. 815–824.
[17] R.K. Amplayo, S. Lee, M. Song, Incorporating product description to sentiment topic models for improved aspect-based sentiment analysis, Information Sciences 454 (2018) 200–215.
[18] F. Li, M. Huang, X. Zhu, Sentiment analysis with global topics and local dependency, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 24, 2010.
[19] M.M. Rahman, H. Wang, Hidden topic sentiment model, in: Proceedings of the 25th international conference on world wide web, 2016, pp. 155–165.
[20] S. Xiong, K. Wang, D. Ji, B. Wang, A short text sentiment-topic model for product reviews, Neurocomputing 297 (2018) 94–102.
[21] A. Sengupta, S. Roy, G. Ranjan, Ljst: A semi-supervised joint sentiment-topic model for short texts, SN Computer Science 2 (4) (2021) 1–16.

[22] X. Chen, S. Jia, Y. Xiang, A review: Knowledge reasoning over knowledge graph, Expert Systems with Applications 141 (2020) 112948.
[23] C. Wang, B. Wang, B. Huang, S. Song, Z. Li, Fastsgg: Efficient social graph generation using a degree distribution generation model, in: 2021 IEEE 37th International Conference on Data Engineering (ICDE), IEEE, 2021, pp. 564–575.
[24] N. Shanavas, H. Wang, Z. Lin, G. Hawe, Knowledge-driven graph similarity for text classification, International Journal of Machine Learning and Cybernetics 12 (4) (2021) 1067–1081.
[25] T. Zhou, K.M. Law, Semantic relatedness enhanced graph network for aspect category sentiment analysis, Expert Systems with Applications 195 (2022) 116560.
[26] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, T. Tan, Session-based recommendation with graph neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 346–353.
[27] F. Chen, Y.-C. Wang, B. Wang, C.-C.J. Kuo, Graph representation learning: A survey, APSIPA Transactions on Signal and Information Processing 9 (2020).
[28] P. Goyal, E. Ferrara, Graph embedding techniques, applications, and performance: A survey, Knowledge-Based Systems 151 (2018) 78–94.
[29] A. Ahmed, N. Shervashidze, S. Narayanamurthy, V. Josifovski, A.J. Smola, Distributed large-scale natural graph factorization, in: Proceedings of the 22nd international conference on World Wide Web, 2013, pp. 37–48.
[30] B. Perozzi, R. Al-Rfou, S. Skiena, Deepwalk: Online learning of social representations, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014, pp. 701–710.
[31] A. Grover, J. Leskovec, node2vec: Scalable feature learning for networks, in: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016, pp. 855–864.
[32] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, arXiv preprint arXiv:1609.02907 (2016).
[33] H. Gao, Z. Wang, S. Ji, Large-scale learnable graph convolutional networks, in: in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 1416–1424.
[34] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, Q. Mei, Line: Large-scale information network embedding, in: Proceedings of the 24th international conference on world wide web, 2015, pp. 1067–1077.
[35] S. Gao, X. Li, Z. Yu, Y. Qin, Y. Zhang, Combining paper cooperative network and topic model for expert topic analysis and extraction, Neurocomputing 257 (2017) 136–143.
[36] Q. Long, Y. Jin, G. Song, Y. Li, W. Lin, Graph structural-topic neural network, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 1065–1073.
[37] N. Van Linh, T.X. Bach, K. Than, A graph convolutional topic model for short and noisy text streams, Neurocomputing 468 (2022) 345–359.
[38] X. Bresson, T. Laurent, Residual gated graph convnets, arXiv preprint arXiv:1711.07553 (2017).
[39] J. Chen, T. Ma, C. Xiao, Fastgcn: fast learning with graph convolutional networks via importance sampling, arXiv preprint arXiv:1801.10247 (2018).
[40] W. Huang, T. Zhang, Y. Rong, J. Huang, Adaptive sampling towards fast graph representation learning, Advances in neural information processing systems 31 (2018).
[41] H. Jiang, R. Zhou, L. Zhang, H. Wang, Y. Zhang, A topic model based on poisson decomposition, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1489–1498.
[42] S. Mukherjee, A. Awadallah, Uncertainty-aware self-training for few-shot text classification, Advances in Neural Information Processing Systems 33 (2020) 21199–21212.
[43] P. Qian, C. Xi, M. Xu, Y. Jiang, K.-H. Su, S. Wang, R.F. Muzic Jr, Ssc-eke: semi-supervised classification with extensive knowledge exploitation, Information Sciences 422 (2018) 51–76.
[44] W. Lin, Z. Gao, B. Li, Shoestring: Graph-based semi-supervised classification with severely limited labeled data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4174–4182.
[45] J. Shen, W. Qiu, Y. Meng, J. Shang, X. Ren, J. Han, Taxoclass: Hierarchical multi-label text classification using only class names, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 4239–4249.
[46] Y. Meng, J. Shen, C. Zhang, J. Han, Weakly-supervised neural text classification, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp. 983–992.
[47] Q. Jiang, L. Chen, R. Xu, X. Ao, M. Yang, A challenge dataset and effective models for aspect-based sentiment analysis, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 6280–6285.
[48] C. Lin, Y. He, R. Everson, A comparative study of bayesian models for unsupervised sentiment detection, in: Proceedings of the fourteenth conference on computational natural language learning, 2010, pp. 144–152.
[49] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: Proceedings of the 22nd international conference on World Wide Web, 2013, pp. 1445–1456.