



# SR-CoMbEr: Heterogeneous Network Embedding Using Community Multi-view Enhanced Graph Convolutional Network for Automating Systematic Reviews

Eric W. Lee<sup>(✉)</sup>  and Joyce C. Ho 

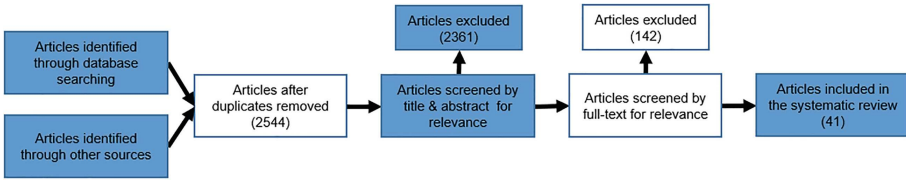
Emory University, Atlanta, GA, USA  
{ewlee4, joyce.c.ho}@emory.edu

**Abstract.** Systematic reviews (SRs) are a crucial component of evidence-based clinical practice. Unfortunately, SRs are labor-intensive and unscalable with the exponential growth in literature. Automating evidence synthesis using machine learning models has been proposed but solely focuses on the text and ignores additional features like citation information. Recent work demonstrated that citation embeddings can outperform the text itself, suggesting that better network representation may expedite SRs. Yet, how to utilize the rich information in heterogeneous information networks (HIN) for network embeddings is understudied. Existing HIN models fail to produce a high-quality embedding compared to simply running state-of-the-art homogeneous network models. To address existing HIN model limitations, we propose SR-CoMbEr, a community-based multi-view graph convolutional network for learning better embeddings for evidence synthesis. Our model automatically discovers article communities to learn robust embeddings that simultaneously encapsulate the rich semantics in HINs. We demonstrate the effectiveness of our model to automate 15 SRs.

**Keywords:** Systematic review · Network embedding · Heterogeneous information network · Multi-view learning · Graph convolution network

## 1 Introduction

Systematic reviews (SRs) serve as a cornerstone of evidence-based medicine and bridge the research-to-practice gap by ensuring all the available evidence is accessible to decision-makers. An excellent SR carefully synthesizes individual studies such as clinical trial results to guide and inform clinical practice. As a motivating example, a SR was used to synthesize findings from randomized intervention studies to determine the impact of angiotensin-converting-enzyme (ACE) inhibitors for treating high blood pressure [7]. As a result, ACE inhibitors now are commonly prescribed to treat hypertension, heart failure, and various other heart conditions. Unfortunately, conducting a SR is an extremely time-consuming and complex task [12]. Established methodologies for performing a



**Fig. 1.** A simplified illustration of the SR screening process using “ACEInhibitors” from Cohen [7] dataset.

SR require a comprehensive search to identify all the relevant studies for inclusion [5]. Yet these broad searches yield imprecise search results (e.g., <2% relevant documents). Figure 1 provides an example of the laborious citation screening process for ACE inhibitors. Only 1.19% of the articles were selected for full-text review based on the title and abstract of which 1.61% were included (i.e., analyzed and evaluated) in the actual review itself. Thus current estimates for the average time to conduct a SR is 67 weeks from registration to publication [3]. Clearly, this process is unsustainable nor scalable, especially given the exponential growth of biomedical literature [2].

Given the importance of SR and the labor-intensive work it entails, research on machine learning and text mining methods to automate the evidence synthesis while maintaining the rigor of a traditional SR have been proposed [28]. In particular, semi-automation can help speed up the screening process, an extremely tedious endeavor due to a large number of articles [25]. The standard methodology for automating the screening process focuses predominately on the text itself using representations like bag-of-words or word embeddings [15, 19, 24]. Yet, recent work demonstrated that the rich citation structure can be utilized to improve the screening process [20]. Their work used a homogeneous network embedding technique, LINE, to learn the citation network representations and these representations were able to outperform the text itself on 10 of the 15 SRs. These promising results suggest that better network representation may expedite evidence synthesis.

Citation networks can be represented as a graph structure that includes articles (nodes) and references (edges). This representation is used across many application domains including social networks, the world wide web, and knowledge graphs. As real-world networks can be huge and complex, it is difficult to directly analyze the graph, thus learning meaningful low-dimensional vectors of the nodes and edges, or network embeddings have been proposed while preserving the features of the network [21]. Recently, there has been an emergence of deep learning-based models such as graph neural networks (GNN) to learn the network embeddings [11, 22, 31]. One popular method is Graph Convolutional Network (GCN) [17] which can efficiently learn the structural dependencies through convolutional operations on the graph. However, GCN is designed for a homogeneous network, whereas the biomedical citation graph contains multiple objects

(nodes) and link types (edges) including author information, venue information, and Medical Subject Headings (MeSH) terms that are used for indexing articles.

Since many real-world networks are heterogeneous information networks (HIN) with multiple objects and link types, several variations of GNN and GCN models have been proposed for HIN embeddings. However, existing models have focused on preserving the meta-path structure (i.e., the path with various object types and edge types that captures the semantics of the network) by transforming the HIN into several homogeneous networks to learn the representations [9, 33, 42]. Unfortunately, the defined meta-path impacts the embedding quality. Thus, ie-HGCN [40] automatically evaluates all possible meta-paths and projects the representations of different types of neighbor objects into a common semantic space. Yet, ie-HGCN is susceptible to noise in the graph.

We propose SR-CoMbEr, a **Community Multi-view based Enhanced Graph Convolutional Network for Systematic Review**. SR-CoMbEr constructs multiple local GCNs, each centered around a community. To learn from the different object and link types, each community adopts a multi-view approach where a view-specific representation is learned to capture the complex structure information for each relation type. Moreover, we pose the multiple community GCN aggregation problem as a multi-modal problem to yield a robust final embedding that reflects the different community representations. Our main contributions of this work are:

- We pose the problem of HIN representation as a multi-view learning problem to avoid specification of the meta-path while automatically capturing the network semantics.
- We propose an innovative multiple, community-based multi-view GCN to capture the structural heterogeneity that is useful for downstream tasks.
- We conduct extensive experiments on SR screening to demonstrate the superior performance of SR-CoMbEr over HIN baselines.

## 2 Preliminaries

In this section, we introduce the *heterogeneous information network*, or HIN, and *Graph Convolutional Network*, or GCN, a state-of-the-art network embedding model.

### 2.1 Heterogeneous Information Network

A HIN contains multiple types of objects and links. Formally, such a network is defined as follows.

**Definition 1. HIN.** A HIN is defined as  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \phi, \psi)$ , where  $\mathcal{V}$  is the set of objects,  $\mathcal{E}$  is the set of links,  $\phi$  is the object type mapping function, and  $\psi$  is the link mapping function.  $\phi$  is defined as  $\phi : \mathcal{V} \rightarrow \mathcal{A}$ , and  $\psi$  is defined as  $\psi : \mathcal{E} \rightarrow \mathcal{R}$ .  $\mathcal{A}$  and  $\mathcal{R}$  denotes predefined object and link types respectively where  $|\mathcal{A}| + |\mathcal{R}| > 2$ .

A homogeneous network contains a single object and relation type such as a social network with User (U) as an object type and a single type of link  $U - U$ . On the other hand, HIN contains multiple types of objects such as a bibliographic network which has four types of objects (i.e., Author (A), Paper (P), Venue (V), and MeSH terms (M) and three link types,  $A - P$ ,  $P - V$ , and  $P - M$ ).

## 2.2 Graph Convolutional Networks

GCNs have been extensively studied and used for a wide range of tasks (see [17] for a survey). Formally, GCNs can be defined as follows. Suppose  $H^k$  is the feature representation of the  $k$ -th layer in GCN, the propagation becomes

$$H^k = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{k-1} W^k) \quad (1)$$

where  $\tilde{A} = A + I \in \mathbf{R}^{N \times N}$  is the adjacency matrix  $A$  with a self connection.  $\tilde{D}$  is the degree matrix of  $\tilde{A}$  which is formally defined as  $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ . And  $W^k$  is a trainable weight matrix. As shown in Eq. (1), the convolution operation is determined by the given graph structure and GCN only learns the node-wise linear transform  $H^{k-1} W^k$ . Thus, the convolution layer can be interpreted as the composition of a fixed convolution followed by an activation function  $\sigma$  on the graph after the node-wise linear transformation.

## 3 Related Works

Methods for semi-automating the citation screening step of SRs have been widely studied [28]. Most of these models use bag-of-words and their combinations as input representations to a supervised learning model (e.g., support vector machine or random forest) [7, 15]. For example, Cohen *et al.* [6] proposed to use uni-grams and bi-grams to treat each of them as a single word, and Bannach-Brown *et al.* [1] used tri-gram and NLP tagger prior to extracting uni-grams.

However, articles contain rich information besides the text, such as citations, author, venue, and keywords. This information can be captured using a HIN where network embeddings can serve as the article representation. Several HIN network embedding methods have been proposed. Existing work focuses on preserving the meta-path structure which contains the semantic information of the graph. For example, ESIM [33] uses multiple user-defined meta-path to learn representations in the user-preferred embedding space, and metapath2vec [9] is a skip-gram model that uses meta-path based random walk. Some works extend Graph Neural Networks (GNNs) for modeling HIN. For example, HAN [38] transforms the given HIN into a homogeneous network based on the meta-path and uses GNN based on hierarchical attention.

However, these models require manually selected meta-path or only accept one meta-path which may cause an information loss by not capturing all meaningful relations. Thus, some recent works proposed learning the meta-path. GTN [41] learns the meta-path to generate multiple new graphs based on the defined meta-path to apply GCN, and ie-HGCN [40] learns the weights of the

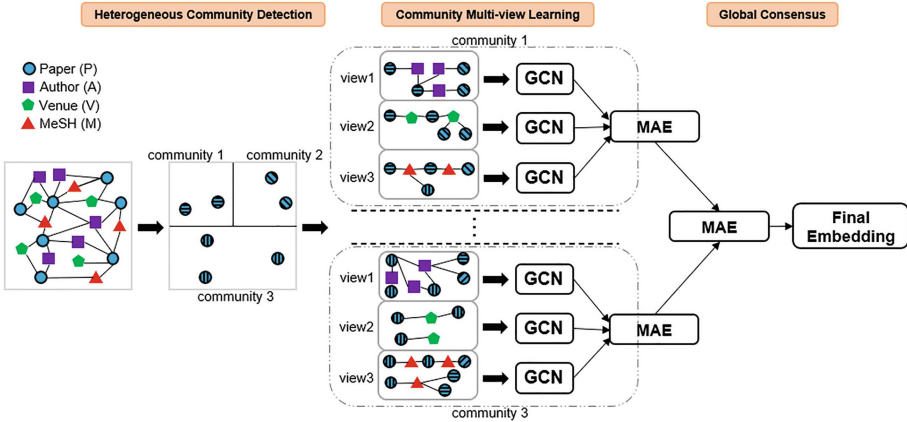
meta-paths to select the best meta-path for their model. While HIN embedding methods are proposed by enhancing GCN by learning meta-path, some research attempts to use multi-view learning for HIN embedding. For example, Zhang *et al.* [43] propose to use a fusion of multiple GCNs modalities of brain images in relationship prediction, and Ma *et al.* [23] uses multi-view graph auto-encoder to capture the similarities of drug features.

## 4 SR-CoMbEr

SR-CoMbEr is inspired by the multiple-filtering local GCN model [39], which constructs multiple local versions of a homogeneous network to capture different aspects of the node attributes while providing robustness to noise. Yet, the local versions of the multiple GCN approach may fail to capture the complex neighborhood structure when solely focusing on a homogeneous network. Moreover, the model can be sensitive to the number of local filters. We address these limitations using three parts: (1) automatic identification of communities in HIN, (2) community multi-view learning to capture information from each link type, and (3) global consensus across the communities. Figure 2 depicts SR-CoMbEr’s overall architecture, where the goal is to learn the representation of the target object  $\alpha$  (i.e., circle node (P)).

### 4.1 Heterogeneous Community Detection

The ability to capture the neighborhood information is a crucial aspect of ensuring the quality of the network embedding. Many network embedding methods use random walks to capture the neighborhoods before passing them to a deep learning model. For example, the multiple-filtering local GCN model [39] uses random walk to construct  $\mathcal{M}$  local networks are constructed. However, sampling of a single link type may not encapsulate the community structure via other link types while sampling multiple links may not be sufficient to capture the complicated structure [42]. However, utilizing the entire HIN can pose computational problems for large networks as well as limit their generalizability to unseen data [39]. Instead, we propose to utilize the community structure ubiquitous in networks, where a group of nodes exhibits more intra-connections than inter-connections with external nodes [10], to determine the construction of the local networks. Given a set of communities, a random walk is initiated using the nodes belonging to the community. Thus each local GCN version learns a better local embedding by integrating information found in the community structure. It is important to note that SR-CoMbEr does not restrict the random walk to just links between community nodes, therefore the local network may contain neighborhood information of nodes outside the community. Moreover, since a node may be part of multiple communities, the combination of multiple local GCNs will thereby reflect different neighborhood information for the same object.



**Fig. 2.** The framework overview of the SR-CoMbEr. The input network is a toy example of a PubMed Network which contains four node types and three edge types. Four node types are Paper (P), Author (A), Venue (V), and MeSH Terms (M), and three edge types are P – A, P – V, and P – M. The target node is set to P which is used for the node classification task.

The community-based focus of each local GCN lends itself naturally to automatic detection of the “optimal” number of local filters,  $\mathcal{M}$ . While there are many types of community detection methods including clustering-based methods [26] and modularity-based methods [27], many of these models are developed for the homogeneous setting. Instead, SR-CoMbEr uses Tucker decomposition [35], a popular tensor factorization model, to identify the community structure and the number of optimal filters in the HIN setting. Tucker decomposition can be viewed as a generalization of singular value decomposition (SVD) which can detect communities in homogeneous networks [30]. The HIN tensor,  $\mathcal{X}$ , is a higher-order tensor where each object type serves as a mode of the tensor and the entries in the tensor capture the status of the links between the different modes of the tensor. For Fig. 2, a paper by author by venue by term tensor (4-mode tensor), can be constructed where each element captures who authored a paper, where it was published, and what terms were present in that paper. Thus, the tensor succinctly encapsulates the relations between different object types.

Formally, for a 3<sup>rd</sup> order tensor,  $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ , Tucker decomposition approximates the tensor into a core tensor,  $\mathcal{H} \in \mathbb{R}^{P \times Q \times S}$  multiplied by a factor matrices along each mode,  $A \in \mathbb{R}^{I \times P}$ ,  $B \in \mathbb{R}^{J \times Q}$ ,  $C \in \mathbb{R}^{K \times S}$ :

$$\mathcal{X} \approx \mathcal{H} \times_1 A \times_2 B \times_3 C. \tag{2}$$

The core tensor,  $\mathcal{H}$  captures the level of interactions between the different components, and the factor matrices,  $A, B, C$ , are often assumed to be column-wise orthonormal. We note that Tucker decomposition generalizes to any  $N$ -mode tensor, does not impose column-wise orthonormal factor matrices nor does the

core tensor have decreasing Frobenius norm along each matrix slice. The general Tucker properties deviate from the SVD assumptions but can be integrated through the algorithmic choice for computing the decomposition. In addition, the column rank of each factor matrix can be different (i.e.,  $P \neq Q \neq S$ ) in the Tucker decomposition. We refer the reader to [18, 29] for additional details.

Since each local filter encapsulates a community, the column rank of each factor matrix is set to be the same,  $P = Q = S$ . To compute the Tucker decomposition, we use the higher-order orthogonal iteration (HOOI) algorithm as it is one of the more efficient techniques. HOOI uses SVD to compute the orthonormal basis of each factor matrix [8]. Moreover, the resulting core tensor and factor matrices can be seen as the generalized counterparts of the matrix SVD. Thus, the superdiagonal entries of the core tensor ( $H_{iii}, \forall i \in [1, R]$ ) is comparable to the singular values of SVD (i.e., diagonal entries in  $\Sigma$ ). As a result, the number of communities can be calculated as the point in which the superdiagonal values converge, similar in fashion to using the  $\Sigma$  matrix in SVD to find the number of communities in a homogeneous network [30]. This eliminates the need for the user to grid search the number of filters  $\mathcal{M}$ .

The next step is to identify the nodes that belong to each community. Without loss of generality, we assume that the target object,  $\alpha$ , corresponds to the first mode of the tensor. Each object can then be represented in a low-dimensional vector space (i.e.,  $P \ll I$ ) using the row vectors of the corresponding factor matrix  $A$ . Spectral clustering is performed on  $A$  to identify the community members using  $\mathcal{M}$  for each node in the target object  $\alpha$ . For simplicity of implementation, SR-CoMbEr uses the k-means algorithm to generate a hard cluster assignment but the framework can use any spectral clustering method. The graph for each community (local) filter is then obtained by performing a fixed-size random walk starting with only nodes within the community. Note that the community filters can contain not just nodes within the same community but also other nodes that are connected during the random walk process. The entire community detection process is summarized in Algorithm 1.

## 4.2 Community Multi-view Learning

Since random walk of  $\mathcal{G}$  directly may fail to capture the complex structure, SR-CoMbEr treats each link type as a different view of the network. For each link type containing the target object  $\alpha$ , a view of the community is created by performing the fixed-size random walk using only this link type. For each community GCN  $m$ , a view is constructed from each link type thus yielding  $|\mathcal{R}|$  different representations,  $X_1^m, \dots, X_{|\mathcal{R}|}^m$ . As an example, three views are constructed for Fig. 2 with a different link type (e.g., P-A, P-V, P-M). Thus, rather than having a single community GCN, each community will have multiple view-specific filters of the network.

Although each view contains a single link type, GCN still cannot be applied directly because the neighbors of an object are of different types. Moreover, the adjacency matrix is not a square matrix and thus cannot be fed into Equation (1),

**Algorithm 1:** Heterogeneous Community Detection in SR-CoMbEr.**Input:** Graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \phi, \psi)$ ,  $\phi : \mathcal{V} \rightarrow \mathcal{A}$ ,  $\psi : \mathcal{E} \rightarrow \mathcal{R}$ **Output:** Number of filters  $\mathcal{M}$ , Communities  $C_1, \dots, C_M$ 

- 1 Construct tensor  $\mathcal{X}$  from  $\mathcal{G}$ ;
- 2 Compute  $\mathcal{X} \approx \mathcal{H} \times_1 A \times_2 B \times_3 C$  using HOOI;
- 3 Set  $\mathcal{M}$  based on convergence of values in the superdiagonal entries of  $\mathcal{H}$ ;
- 4 Detect communities of  $\alpha$ ,  $C_1, C_2, \dots, C_M$ , using spectral clustering of  $A$ ;
- 5 **return**  $\mathcal{M}, C_1, C_2, \dots, C_M$ ;

where  $\tilde{A}$  is the square matrix. We thus use the idea of projection, introduced in ie-HGCN [40], to ensure both object types are in the same space. Suppose the view captures the link  $\alpha\text{-}\beta$ , where  $\mathcal{V}^\alpha$  and  $\mathcal{V}^\beta$  represent the set of objects in the  $\alpha$  and  $\beta$  node type, respectively. Let  $A^{\alpha\text{-}\beta} \in \mathbf{R}^{|\mathcal{V}^\alpha| \times |\mathcal{V}^\beta|}$  denote the adjacency matrix between  $\alpha$  and  $\beta$  and the degree matrix  $D^{\alpha\text{-}\beta} = \text{diag}(\sum_j A_{ij}^{\alpha\text{-}\beta}) \in \mathbf{R}^{|\mathcal{V}^\alpha| \times |\mathcal{V}^\alpha|}$ . Every object is then projected into the same space and passed to the GCN:

$$\begin{aligned} \tilde{A}^{\alpha\text{-}\beta} &= (D^{\alpha\text{-}\beta})^{-1} \cdot A^{\alpha\text{-}\beta} \\ X_{\alpha\text{-}\beta} &= \tilde{A}^{\alpha\text{-}\beta} \cdot W^{\alpha\text{-}\beta} \end{aligned} \quad (3)$$

where  $\tilde{A}^{\alpha\text{-}\beta}$  is the row-normalized matrix and  $W^{\alpha\text{-}\beta}$  is the trainable convolution weight matrix of  $\alpha\text{-}\beta$  relation.

The community embedding,  $X^m$ , should capture all the information from the  $|\mathcal{R}|$  views while reducing information redundancy that may be present in the views. Moreover, certain views may learn better representations of the community. Thus, to summarize the different view modalities simultaneously, SR-CoMbEr adopts the multi-modal stacked autoencoder (MAE) [4]. MAE takes multiple input representations, concatenates the input together, and then passes this to an autoencoder to induce a succinct, shared representation from which to reconstruct the original (concatenated) inputs. Formally, the global consensus process for the shared representation in the  $m^{\text{th}}$  community GCN is:

$$H^m = \text{MAE}(X_1^m, X_2^m, \dots, X_{|\mathcal{R}|}^m). \quad (4)$$

### 4.3 Global Consensus

Since each community multi-view GCN representation  $H^m$ , captures community-specific information, the learned representation can differ. We formulate the aggregation of the community multi-view GCN representation as a multi-modal problem. Although the final shared representation can be computed as the average of the community representations, this assumes each community is equivalent. In practice, some community representations are of higher quality and thereby should have higher weights. MAE is used again to learn the final representation across the  $\mathcal{M}$  communities:

$$H = \text{MAE}(H^1, H^2, \dots, H^{\mathcal{M}}) \quad (5)$$



---

**Algorithm 2:** The pseudocode of SR-CoMbEr.

---

**Input:** Graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \phi, \psi)$ ,  $\phi : \mathcal{V} \rightarrow \mathcal{A}$ ,  $\psi : \mathcal{E} \rightarrow \mathcal{R}$   
Number of localized filters  $\mathcal{M}$   
**Output:** Final representation  $H$

- 1 Compute  $\mathcal{M}, C_1, C_2, \dots, C_M$  using Algorithm 1;  
/\* Loop through the communities \*/
- 2 **for**  $i=1, \dots, \mathcal{M}$  **do**  
| /\* Loop through the views \*/
- 3 | **for**  $\alpha - \beta \in \mathcal{R}$  **do**
- 4 | | Run random walk on objects  $\in C_i^\alpha$  and  $\in C_i^\beta$ ;
- 5 | | Compute  $X_{\alpha-\beta}$  according to Eq. (3);
- 6 | **end**
- 7 | Compute  $H^i$  according to Eq. (4);
- 8 **end**
- 9 Compute loss and update parameters;
- 10 **return**  $H$  according to Eq. (5);

---

The final embedding representation,  $H$ , is then used for a variety of tasks such as classification, clustering, etc., where the loss function is tailored towards the specific task. For example, in a multi-class node classification task,  $H$  is passed to a fully connected layer with softmax activation, and the loss is defined as the cross-entropy over the object type. The weights are then learned using stochastic gradient descent with backpropagation. Algorithm 2 shows the overall training procedure of SR-CoMbEr.

## 5 Experimental Design

### 5.1 Dataset

We evaluate our model on the publicly available dataset provided by Cohen *et al.* [7]. The dataset includes 15 SRs (or topics) concerning different drug efficacies which were performed by members of evidence-based practice centers (EPCs). In the dataset, each SR topic contains a set of PubMed article identifiers (PMID) and their associated title/abstract screening status (i.e., whether or not the article passed the title/abstract screening stage). The PMID allows us to retrieve the metadata (citation, author, venue, and MeSH terms) from the PubMed database. There exist other SR datasets [32], however, the dataset does not contain the PMID. We extract a subset of articles from the PubMed database using Entrez API<sup>1</sup>. Including all the articles from the Cohen dataset and using Entrez API, we trace articles up to 2-hops based on the citation information and retrieve about 7.6M articles with the meta-data including author, venue, and MeSH terms. The number of articles screened ranged from 310 (Antihistamines) to 3465 (Statins)

<sup>1</sup> <https://www.ncbi.nlm.nih.gov/books/NBK25501/>.

**Table 1.** Comparison of baseline characteristics. The \* symbol next to the model name denotes a homogeneous network model. The columns MP, SS, and MVF represent meta-path specification, subgraph sampling, and multi-view fusion, respectively.

	MP	SS	MVF	Module	Supervision
LINE*	✗	✗	✗	Skip-gram	✗
GCN*	✗	✗	✗	GCN	✓
HAN	✓	✗	✗	Transformer	✓
GAHNE	✗	✓	✓	GCN	✓
ie-HGCN	✗	✓	✓	GCN	✓
SR-CoMbEr	✗	✓	✓	GCN	✓

with anywhere from 2.07% (SkeletalMuscleRelaxants) to 32.49% (Triptans) passing the abstract screening process. This demonstrates a relatively large degree of imbalance.

## 5.2 Baselines

We compare with five baselines spanning both homogeneous and HIN embedding methods in the SR task. Table 1 compares the characteristics of baseline models.

- **LINE** [34]. A conventional network embedding method that is using first- and second-proximity. Since it is designed for a homogeneous network, we transform the HIN by considering collapsing the object and link types as a single type and use LINE to learn the representation of the whole HIN.
- **GCN** [17]. A semi-supervised graph convolutional network that is designed for a homogeneous network. Similar to LINE, we ignore the heterogeneity of the network and collapse it into a homogeneous network..
- **HAN** [38]. A model to learn representations for HIN. It transforms the HIN into several homogeneous sub-networks by user-defined meta-paths. For object-level aggregation, it uses GAT [36], then uses an attention mechanism to fuse object representations from each sub-networks.
- **GAHNE** [21]. A model to learn representations for HIN. It converts the network into a series of homogeneous sub-networks to capture the semantic information. Then an aggregation mechanism fuses the sub-networks with supplemental information from the whole network.
- **ie-HGCN** [40]. A HIN embedding model that evaluates all possible meta-paths and projects the representations of different types of neighbor objects into a common semantic space using object- and type-level aggregation.

## 5.3 Evaluation Metrics

The recent trend for evaluating a SR task uses the area under the receiver operating curve (AUC) for predicting whether or not the abstract was screened or not to report the results [6, 24]. Thus we evaluate the models using AUC.

## 5.4 Implementation Details

Our method is implemented in Keras and the source code is publicly available<sup>2</sup>. The source codes of the other baselines are provided by their authors and are implemented in either PyTorch or TensorFlow. All experiments are conducted on a machine with 1 Nvidia GeForce GTX 1080Ti and 11GB GPU memory. For each SR task, we randomly split the articles in the SR into train-validation-test as 50%–25%–25%, and use the validation set for the hyperparameter tuning. Articles not in the target SR task are marked as irrelevant in the training process.

For the baseline models, we adopt the same hyperparameter settings introduced in their respective papers. For LINE [34], we use a dimension of 128 for each first- and second-order proximity resulting a dimension of 256 for the final embedding. For GCN [17], we use the learning rate of 0.01, the dropout rate of 0.5, and the L2 penalty weight decay of 0.0005. For HAN [38], the number of attention heads is set to 8, and the meta-paths PAP, PMP, and APVPA are used (P: Paper, A: Author, M: MeSH terms, and V: Venue). For GAHNE [21], we used a learning rate of 0.005, a dropout of 0.5, an L2 penalty of 0.001, and a dimension of 128. For ie-HGCN [40], the number of layers is set to 5, and the dimension for the four hidden layers starting from the second layer is set to 64, 32, 16, and 8. For SR-CoMbEr, we use  $\mathcal{M} = 12$ , set the random walk length to 20, and the embedding dimension to 128. The Adam optimizer [16] is used with a learning rate of 0.01 and all parameters are initialized randomly. Dropout is used for all layers except the output layer with a dropout rate of 0.5.

## 6 Experimental Results

### 6.1 Systematic Review

The AUC on the Cohen dataset is reported in Table 2 for each SR. The best results are bolded and the second-best results are underlined. The results show that HIN embedding outperforms homogeneous network embedding (LINE and GCN). This demonstrates citation information and other node types (author, venue, and MeSH terms) help to improve the performance of the SR task.

From the table, we observe SR-CoMbEr outperforms all other baselines from 0.002 to 0.018 by comparing with the second-best AUC score. This indicates the importance of effectively modeling the HIN and demonstrates the effectiveness of SR-CoMbEr in the SR task. Between the existing HIN models, HAN shows the limitation of the user-defined meta-path. The results suggest that there are more hidden but important paths that are difficult for users to define. In contrast, the performance between GAHNE and ie-HGCN is similar. GAHNE performs better when there are more papers excluded from the abstract screening process. For example, the “SkeletalMuscleRelaxants” dataset has a total of 1643 articles in the beginning but only 34 articles are selected from the abstract screening which is only 2%. While GAHNE performs better in cases when fewer articles are

<sup>2</sup> <https://github.com/ewhlee/SR-CoMbEr>.

**Table 2.** Performance results (AUC score) for the SR task. The best score for each SR is bolded and the second highest is underlined.

SR	LINE	GCN	HAN	GAHNE	ie-HGCN	SR-CoMbEr
ACEInhibitors	0.622	0.627	0.649	0.662	<u>0.667</u>	<b>0.672</b>
ADHD	0.597	0.605	0.621	0.644	<u>0.646</u>	<b>0.659</b>
Antihistamines	0.541	0.544	0.567	<u>0.588</u>	0.586	<b>0.593</b>
AtypicalAntipsychotics	0.601	0.607	0.617	<u>0.638</u>	0.636	<b>0.641</b>
BetaBlockers	0.629	0.632	0.658	0.671	<u>0.677</u>	<b>0.684</b>
CalciumChannelBlockers	0.636	0.64	0.662	<u>0.67</u>	0.666	<b>0.688</b>
Estrogens	0.577	0.583	0.607	<u>0.629</u>	0.626	<b>0.631</b>
NSAIDs	0.637	0.639	0.662	<u>0.691</u>	0.685	<b>0.697</b>
Opioids	0.632	0.635	0.654	0.667	<u>0.671</u>	<b>0.686</b>
OralHypoglycemics	0.555	0.559	0.582	<u>0.591</u>	0.583	<b>0.598</b>
ProtonPumpInhibitors	0.638	0.641	0.664	0.677	<u>0.681</u>	<b>0.687</b>
SkeletalMuscleRelaxants	0.64	0.643	0.658	0.672	<u>0.677</u>	<b>0.684</b>
Statins	0.606	0.609	0.633	0.653	<u>0.659</u>	<b>0.665</b>
Triptans	0.617	0.624	0.64	0.652	<u>0.66</u>	<b>0.671</b>
UrinaryIncontinence	0.633	0.639	0.658	<u>0.678</u>	0.675	<b>0.683</b>

selected, ie-HGCN performs better in cases when more papers are selected. For example, “AtypicalAntipsychotics” has a total of 1120 articles in the beginning and 363 articles passed the screening which is 32%.

## 6.2 Ablation Study

We assess the importance of each component in SR-CoMbEr for the final embedding. *LMV* is a localized, multi-view model that does not use the heterogeneous community detection component (i.e., Sect. 4.1). Each localized, multi-view filter is subsampled using a random walk of all the nodes in the graph. Then the local representations are aggregated using an average function. *CoAvg* extends *LMV* by using the community detection module to construct the localized, multi-view filters. However, unlike the SR-CoMbEr, it does not use the MAE to learn the shared representation from the community filters (i.e., Eq. (5) is replaced with  $H = \text{AVG}(H^1, H^2, \dots, H^M)$ ). Table 3 summarizes the AUC scores on the test set of the two different multi-view learning techniques on the ACEInhibitors SR task. As shown in the table, incorporating the community information improves the performance (see CoAvg versus LMV). By leveraging the community structure, the embedding model can capture different neighborhood information to learn a better representation. While the overall results suggest that although the performance boost is less compared to the community detection component, MAE is beneficial to automatically learn the weights from each of the community representations for the final embedding.

To better understand the importance of the community detection algorithm, we compared the performance using SVD to identify the communities using just

**Table 3.** Comparison of the AUC score using different community detection algorithms on ACEInhibitors from the SR task.

LMV	CoAvg	CP	SVD	SR-CoMbEr
0.658	0.665	0.668	0.662	<b>0.672</b>

one view of the network [30] and CANDECOMP-PARAFAC (CP), a special case of Tucker decomposition where the core tensor only has values along the superdiagonal entries [18]. For SVD, let  $F \in \mathbb{R}^{m \times n}$  denote the adjacency matrix of the link type with the largest number of nodes and the target node  $\alpha$ . Under SVD,  $F = U\Sigma V^*$ , where  $U \in \mathbb{R}^{m \times p}$ ,  $V \in \mathbb{R}^{n \times p}$  matrix, and  $\Sigma \in \mathbb{R}^{p \times p}$ . Spectral clustering is then performed in a similar fashion using  $\mathcal{M}$  as the number of clusters on the target object,  $\alpha$ , and  $U$  as the low-dimensional embedding. For CP decomposition, the alternating least square method is used to find the leading left singular values [13]. As shown in the table, SR-CoMbEr (using HOOI algorithm) for community detection outperforms other techniques (see CP and SVD). While we identify 12 local filters for SR-CoMbEr using HOOI, SVD identifies 9 and CP identifies 14. This shows the importance of identifying the optimal number of filters as too many or too few filters can degrade the performance.

## 7 Conclusion

In this paper, we propose SR-CoMbEr to learn citation network representations for SRs. To avoid defining the meta-path, we formulate the problem using multi-view learning to automatically capture the semantics of HIN. To encode the structural heterogeneity and neighborhood information, we use community detection and multiple community-based views of the network and fuse the representations to obtain the final representation. We also introduce the use of HOOI to compute the optimal number of filters in concert with community detection. The experiments on 15 SRs show that SR-CoMbEr outperforms several state-of-the-art HIN embedding models.

There are several limitations to our study. First, the improvements in results are not substantially better even after all the extensive modeling. This is typical in SR automation as the evaluation measures may be ill-suited for capturing major improvements due to the dominance of irrelevant documents. Second, our evaluation was limited in the number of topics considered. There are other evaluation resources such as the CLEF eHealth TAR data, clinical outcomes [37], SIGIR 2017 SysRev Query Collection [32], and the SWIFT-review dataset [14] that can be explored for future work. Other promising future directions include the incorporation of the article text as well as the structure of the PubMed HIN.

**Acknowledgements.** We thank the reviewers for their insightful suggestions and comments. This work was supported by the National Science Foundation award IIS-1838200 and IIS-2145411.

## References

1. Bannach-Brown, A., et al.: Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error. *Systematic Rev.* **8**(1), 23 (2019)
2. Bastian, H., Glasziou, P., Chalmers, I.: Seventy-five trials and eleven systematic reviews a day: How will we ever keep up? *PLOS Med.* **7**(9), e1000326 (2010)
3. Borah, R., Brown, A.W., Capers, P.L., Kaiser, K.A.: Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the prospero registry. *BMJ open* **7**(2), e012545 (2017)
4. Cadena, C., Dick, A.R., Reid, I.D.: Multi-modal auto-encoders as joint estimators for robotics scene understanding. In: *Robotics: Science and Systems* (2016)
5. Chandler, J., Churchill, R., Higgins, J., Lasserson, T., Tovey, D., et al.: Methodological standards for the conduct of new cochrane intervention reviews. *Cochrane Collaboration*, SI (2013)
6. Cohen, A.M.: Optimizing feature representation for automated systematic review work prioritization. In: *AMIA Annu. Symp. Proceed.* **2008**, 121–125 (2008). American Medical Informatics Association (2008)
7. Cohen, A.M., Hersh, W.R., Peterson, K., Yen, P.Y.: Reducing workload in systematic review preparation using automated citation classification. *J. Am. Med. Inf. Assoc.* **13**(2), 206–219 (2006)
8. De, L., De-Moor, B., Vandewalle, J.: On the best rank-1 and rank-(r1 r2...rn) approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.* **21**(4), 1324–1342 (2000)
9. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: scalable representation learning for heterogeneous networks. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Halifax, NS, Canada, 13–17 August 2017 (2017)
10. Girvan, M., Newman, M.E.: Community structure in social and biological networks. *Proceedings of the National Academy Of Sciences* (12) (2002)
11. Gori, M., Monfardini, G., Scarselli, F.: A new model for learning in graph domains. In: *Proceedings. 2005 IEEE International Joint Conference on Neural Networks*, 2005. IEEE (2005)
12. Haddaway, N.R., Westgate, M.J.: Predicting the time needed for environmental systematic reviews and systematic maps. *Conserv. Biol.* **33**, 434–443 (2018)
13. Harshman, R.A., et al.: Foundations of the parafac procedure: models and conditions for an “explanatory” multimodal factor analysis (1970)
14. Howard, B.E., et al.: Swift-review: a text-mining workbench for systematic review. *Syst. Control Found. Appl.* **5**(1), 1–16 (2016)
15. Khabsa, M., Elmagarmid, A., Ilyas, I., Hammady, H., Ouzzani, M.: Learning to identify relevant studies for systematic reviews using random forest and external information. *Mach. Learn.* **102**(3), 465–482 (2015). <https://doi.org/10.1007/s10994-015-5535-7>
16. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: *Proceedings of ICLR* (2015)
17. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: *Proceedings of ICLR* (2017)
18. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. *SIAM Rev.* **51**(3), 455–500 (2009)

19. Kontonatsios, G., et al.: A semi-supervised approach using label propagation to support citation screening. *J. Biomed. Inf.* **72**, 67–76 (2017)
20. Lee, E.W., Wallace, B.C., Galaviz, K.I., Ho, J.C.: MMiDaS-AE: multi-modal missing data aware stacked autoencoder for biomedical abstract screening. In: *Proceedings of the ACM Conference on Health, Inference, and Learning* (2020)
21. Li, X., Wen, L., Qian, C., Wang, J.: GAHNE: graph-aggregated heterogeneous network embedding. In: *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE (2020)
22. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.S.: Gated graph sequence neural networks. In: *Proceedings of ICLR* (2016)
23. Ma, T., Xiao, C., Zhou, J., Wang, F.: Drug similarity integration through attentive multi-view graph auto-encoders. In: *Proceedings of the Twenty-Seventh IJCAI 2018*, Stockholm, Sweden (2018)
24. Miwa, M., Thomas, J., O'Mara-Eves, A., Ananiadou, S.: Reducing systematic review workload through certainty-based screening. *J. Biomed. Inf.* **51**, 242–253 (2014)
25. Morris, Z.S., Wooding, S., Grant, J.: The answer is 17 years, what is the question: understanding time lags in translational research. *J. Royal Soc. Med.* **104**(12), 510–520 (2011)
26. Newman, M.E.J.: Detecting community structure in networks. *Eur. Phys. J. B* **38**(2), 321–330 (2004). <https://doi.org/10.1140/epjb/e2004-00124-y>
27. Newman, M.E.: Modularity and community structure in networks. *Proceed. Nat. Acad. Sci.* **103**(23), 8577–8582 (2006)
28. O'Mara-Eves, A., Thomas, J., McNaught, J., Miwa, M., Ananiadou, S.: Using text mining for study identification in systematic reviews: a systematic review of current approaches. *Syst. Rev.* **4**(1), 5 (2015). <https://doi.org/10.1186/2046-4053-4-5>
29. Papalexakis, E.E., Faloutsos, C., Sidiropoulos, N.D.: Tensors for data mining and data fusion: Models, applications, and scalable algorithms. *ACM Trans. Intell. Syst. Technol. (TIST)* **8**(2), 2915921 (2016)
30. Sarkar, S., Dong, A.: Community detection in graphs using singular value decomposition. *Phys. Rev. E* **83**(4), 046114 (2011)
31. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. *IEEE Trans. Neural Netw.* **20**(1), 61–80 (2008)
32. Scells, H., Zuccon, G., Koopman, B., Deacon, A., Azzopardi, L., Geva, S.: A test collection for evaluating retrieval of studies for inclusion in systematic reviews. In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1237–1240 (2017)
33. Shang, J., Qu, M., Liu, J., Kaplan, L.M., Han, J., Peng, J.: Meta-path guided embedding for similarity search in large-scale heterogeneous information networks. *ArXiv preprint* (2016)
34. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: LINE: large-scale information network embedding. In: *Proceedings of the 24th International Conference on WWW*, Florence, Italy, 18–22 May 2015 (2015)
35. Tucker, L.R.: Some mathematical notes on three-mode factor analysis. *Psychometrika* **31**(3), 279–311 (1966). <https://doi.org/10.1007/BF02289464>
36. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: *Proceedings of ICLR* (2018)
37. Wallace, B.C., Trikalinos, T.A., Lau, J., Brodley, C., Schmid, C.H.: Semi-automated screening of biomedical citations for systematic reviews. *BMC Bioinf.* **11**(1), 1–11 (2010)

38. Wang, X., et al.: Heterogeneous graph attention network. In: The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, 13–17 May 2019 (2019)
39. Wanyan, T., Zhang, C., Azad, A., Liang, X., Li, D., Ding, Y.: Attribute2vec: Deep network embedding through multi-filtering GCN. ArXiv preprint (2020)
40. Yang, Y., Guan, Z., Li, J., Huang, J., Zhao, W.: Interpretable and efficient heterogeneous graph convolutional network. ArXiv preprint (2020)
41. Yun, S., Jeong, M., Kim, R., Kang, J., Kim, H.J.: Graph transformer networks. In: Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8–14 Dec 2019, Vancouver, BC, Canada (2019)
42. Zhang, C., Song, D., Huang, C., Swami, A., Chawla, N.V.: Heterogeneous graph neural network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, 4–8 Aug 2019 (2019)
43. Zhang, X., He, L., Chen, K., Luo, Y., Zhou, J., Wang, F.: Multi-view graph convolutional network and its applications on neuroimage analysis for Parkinson’s disease. In: AMIA Annu. Symp. Proceed. **2018**, 1147–1156 (2018). American Medical Informatics Association (2018)