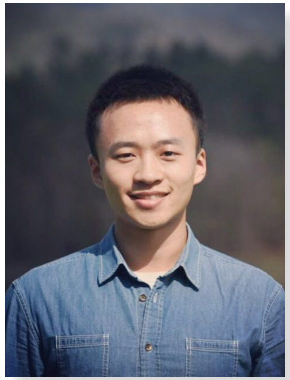
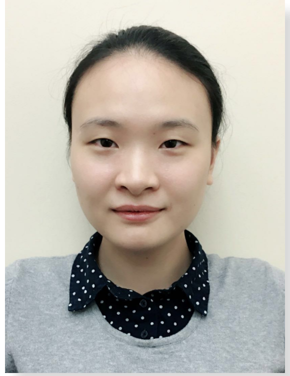
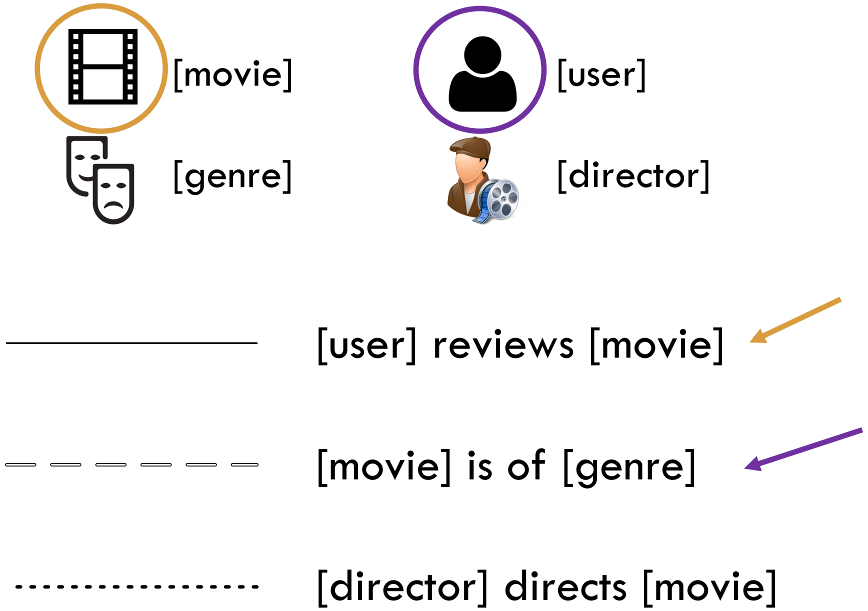
AspEm: Embedding Learning by Aspects in Heterogeneous Information Networks

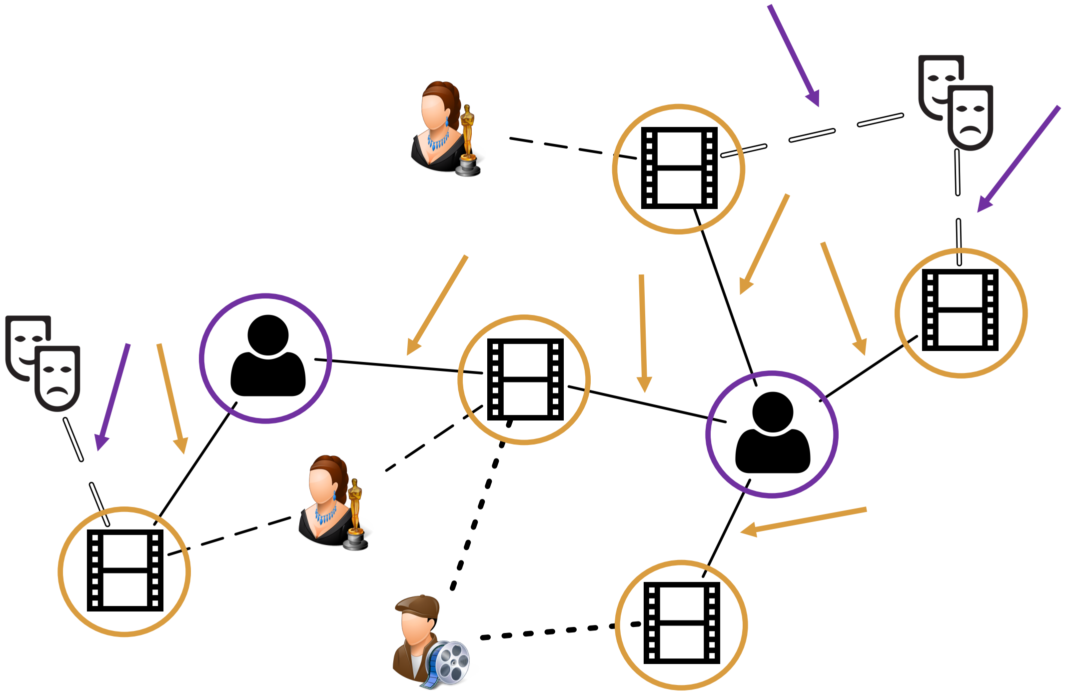
Yu Shi[†], Huan Gui[†‡], Qi Zhu[†], Lance Kaplan[§], Jiawei Han[†]
[†]University of Illinois at Urbana-Champaign (UIUC)
[‡]Facebook Inc. [§]U.S. Army Research Laboratory


In real world applications, objects of different types can have different relations, which form heterogeneous information networks (**HINs**).

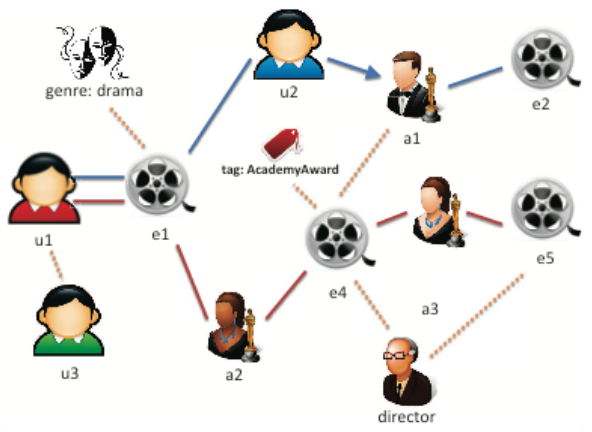
- **Typed** nodes: objects
- **Typed** edges: relations



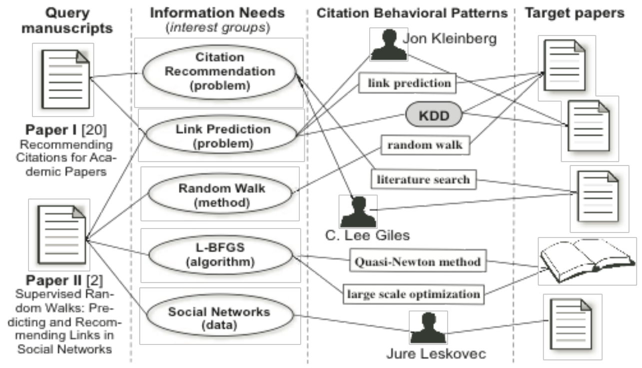


A toy movie reviewing network

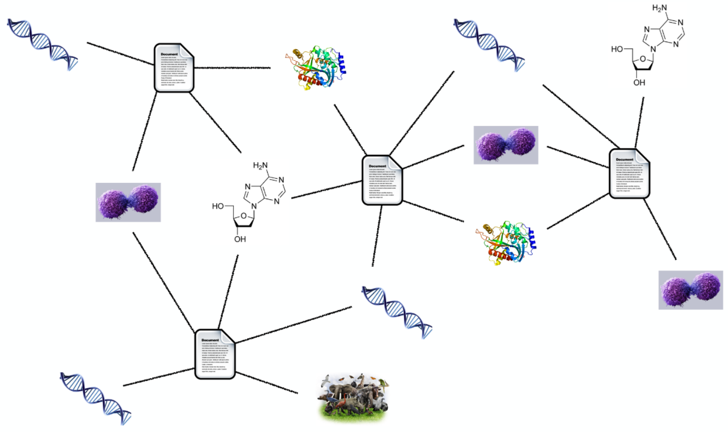
Heterogeneous information networks (HINs) are ubiquitous.



IMDb Network

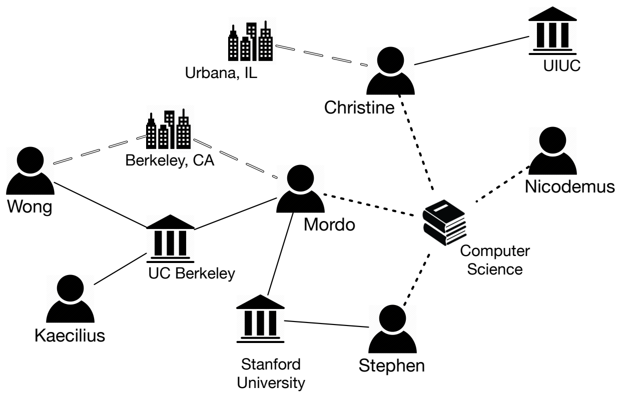


Bibliographical Network

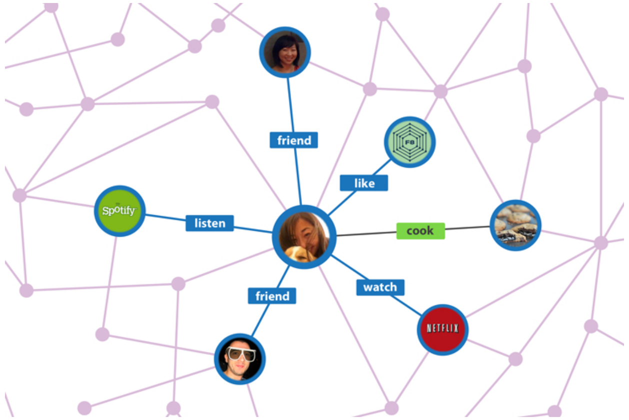


Biomedical Network

Economic Graph



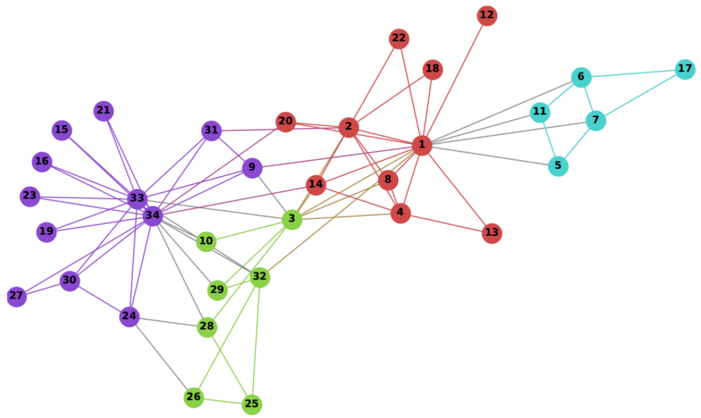
Social Network



Facebook Open Graph

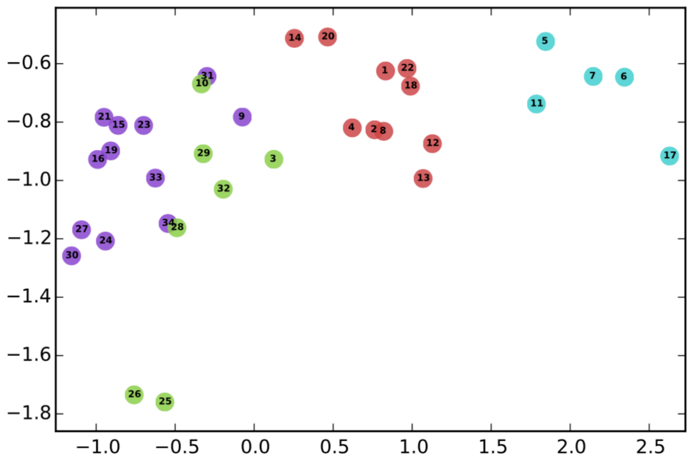
Figures partly from http://xren7.web.engr.illinois.edu/yahoo-dais_award.html, http://www.garrygolden.com/2015/07/06/linkedin-economic-graph-technosolutionism, and http://www.businessinsider.com/explainer-what-exactly-is-the-social-graph-2012-3

Network embedding has been heavily studied recently as a representation learning method.



Input network

Embed – represent nodes by vectors in the embedding space



Embedding space

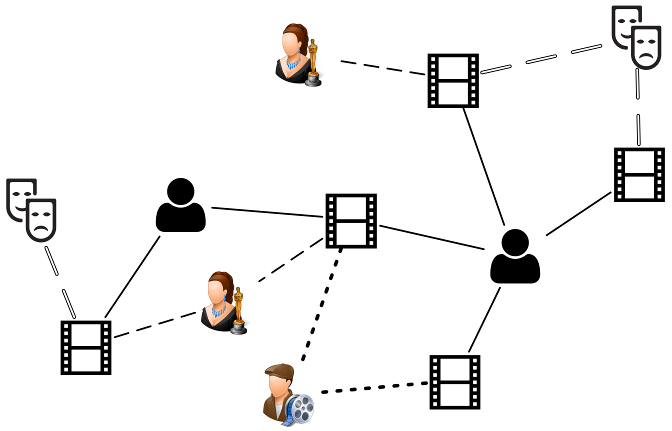
The learned vectors can be used as features in downstream applications

- Node classification
- Link Prediction
- Community detection
- Recommendation
- …

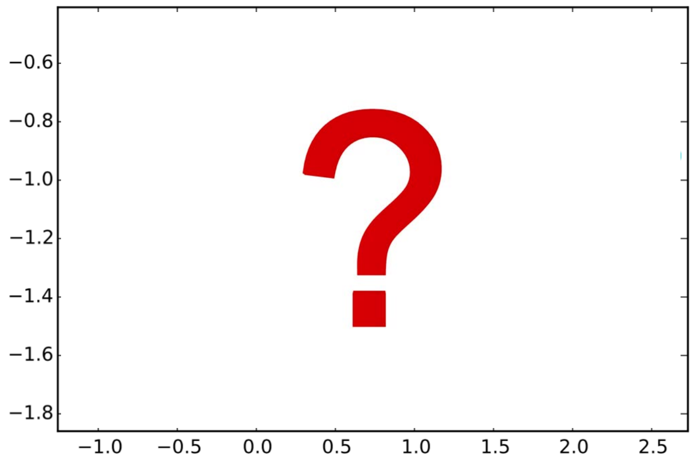
Figures from Perozzi, et al. "Deepwalk: Online learning of social representations." In KDD, 2014.

We are motivated to study the problem of **Embedding Learning in Heterogeneous Information Networks** (HINs).

- What would happen when we embed nodes of various types into the embedding space?



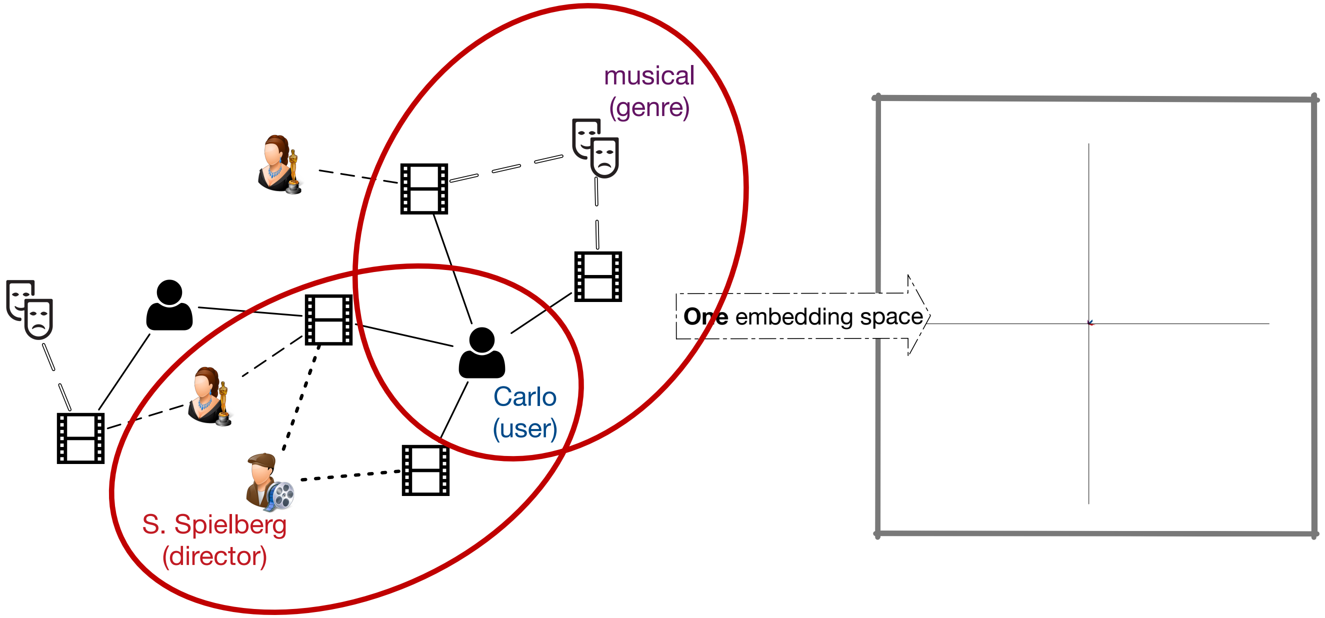
Input network



Embedding space

While the **heterogeneity** in HINs carries rich information, it also poses special **challenges**.

- If we embed all nodes into **the same metric space…**



- **Carlo** likes **musical**, so he should be close to **musical**.
- **Carlo** likes **Spielberg**, so he should be close to **Spielberg**.
- **Spielberg** is semantically dissimilar to **musical**, so their embeddings are far apart.
- As a result, **Carlo** turns out to be **close to neither.**

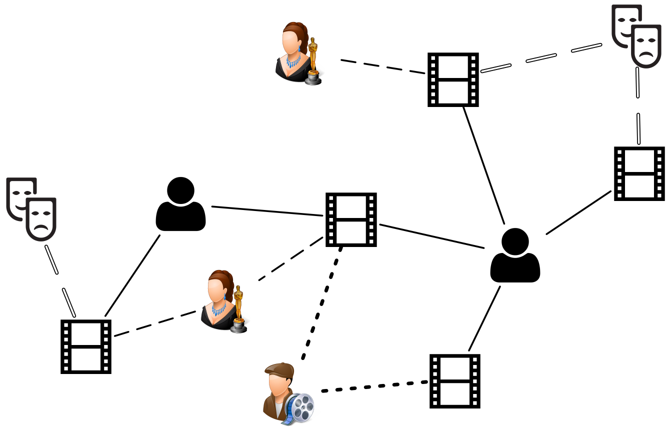
- … we would suffer from **information loss** due to the **incompatibility** among node and edge types.
- It is of interest to develop embedding method that can alleviate this problem
  – i.e., preserve Carlo's preference for both musicals and Spielberg's movies.

We alleviate the problem of information loss due to the incompatibility by embedding **representative aspects** in an HIN,

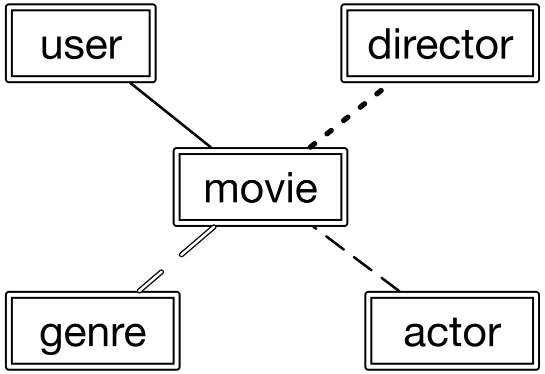
- where a **representative aspect** is a unit representing one **compatible semantic facet** of the HIN,
- and propose the AspEm framework (short for **As**pect **Em**bedding)

The AspEm Framework – Overview

Formally, we define an **aspect** of an HIN by a **connected subgraph** of its schema.



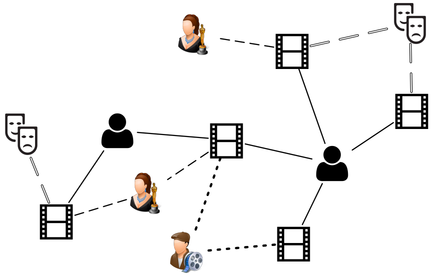
An HIN



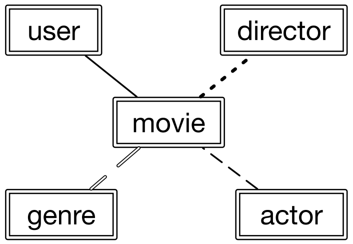
Its **schema**
(an abstraction of the type information)

The AspEm Framework – Overview

Formally, we define an **aspect** of an HIN by a **connected subgraph** of its schema

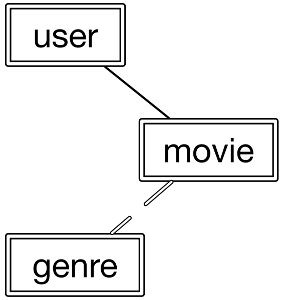


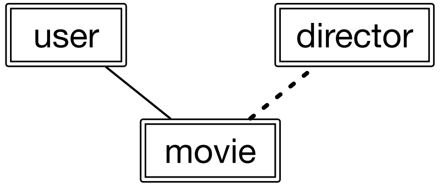
An HIN



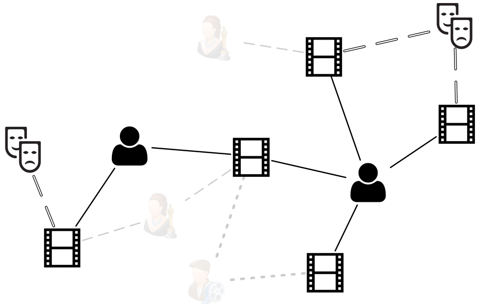
Its schema

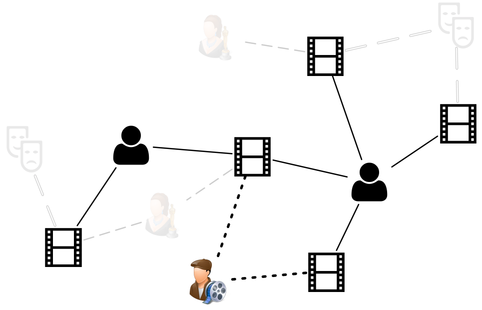
Schema level





Network level

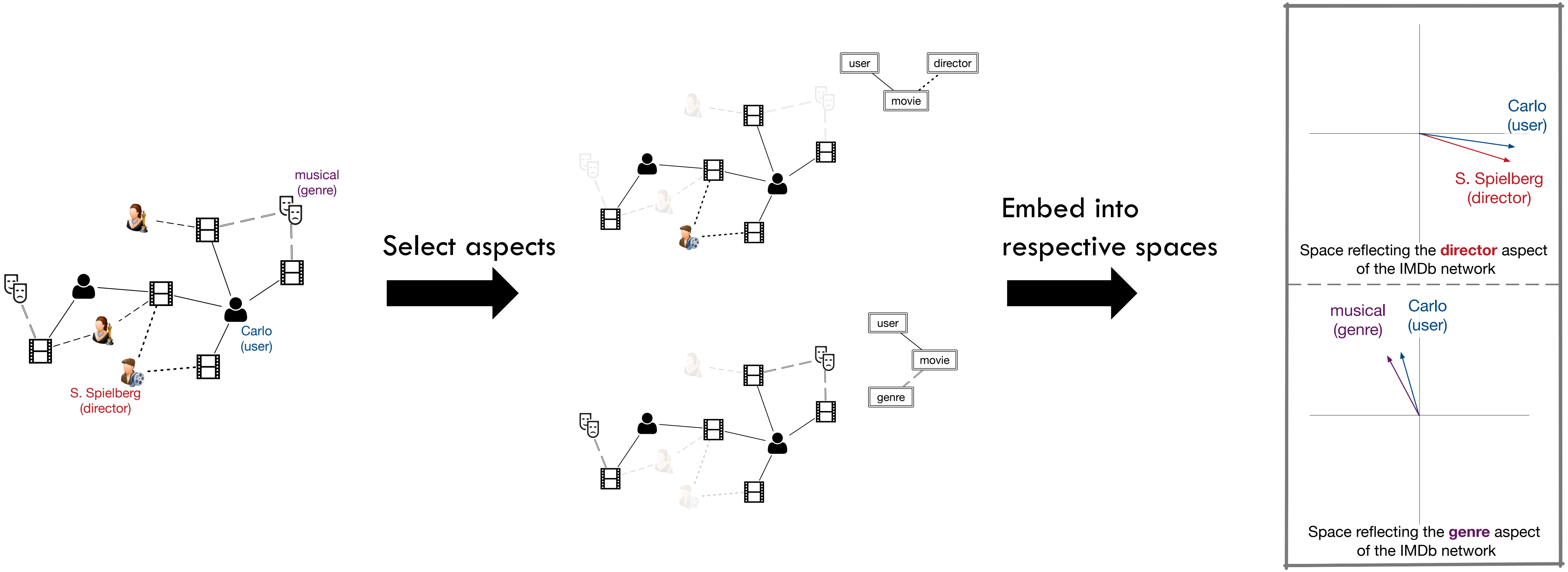




Two example aspects of the HIN

The AspEm Framework – Overview

AspEm first selects a set of aspects, then embeds each aspect into its own metric space



and finally, for any node involved in multiple aspects, concatenates to build its final embedding.

The AspEm Framework – Aspect Selection

How to select aspects?

- When supervision in downstream application is available, one may choose the set of aspects that perform the best in the downstream application.
- When supervision is not available, can we still select a set of representative aspects?
    - Yes.
    - Since a representative aspect corresponds to one **compatible semantic facet**, we should be able to find them by dataset-wide statistics that measures incompatibility.
        - In other words, the selected aspects **should not have incompatible node types and edge types** within themselves.

The AspEm Framework – Aspect Selection

To quantify such incompatibility, we propose the following Jaccard coefficient based measure for aspect incompatibility.

An aspect **A base aspect**

$$\mathrm{Inc}(a) := \sum_{\langle \phi_l, \psi_l, \phi_c, \psi_r, \phi_r \rangle \subseteq a} \mathrm{Inc}(\phi_l \xrightarrow{\psi_l} \phi_c \xrightarrow{\psi_r} \phi_r)$$

The incompatibility of an aspect is aggregated from all its base sub-aspects.

A central node

$$\mathrm{Inc}(\phi_l \xrightarrow{\psi_l} \phi_c \xrightarrow{\psi_r} \phi_r) := \frac{1}{|\phi_c^*|} \sum_{u \in \phi_c^*} \gamma(u)$$

The incompatibility of a base aspect depends on the inconsistency (γ) observed by its central nodes.

The inconsistency (γ) captures the difference in reachability via different edge types.

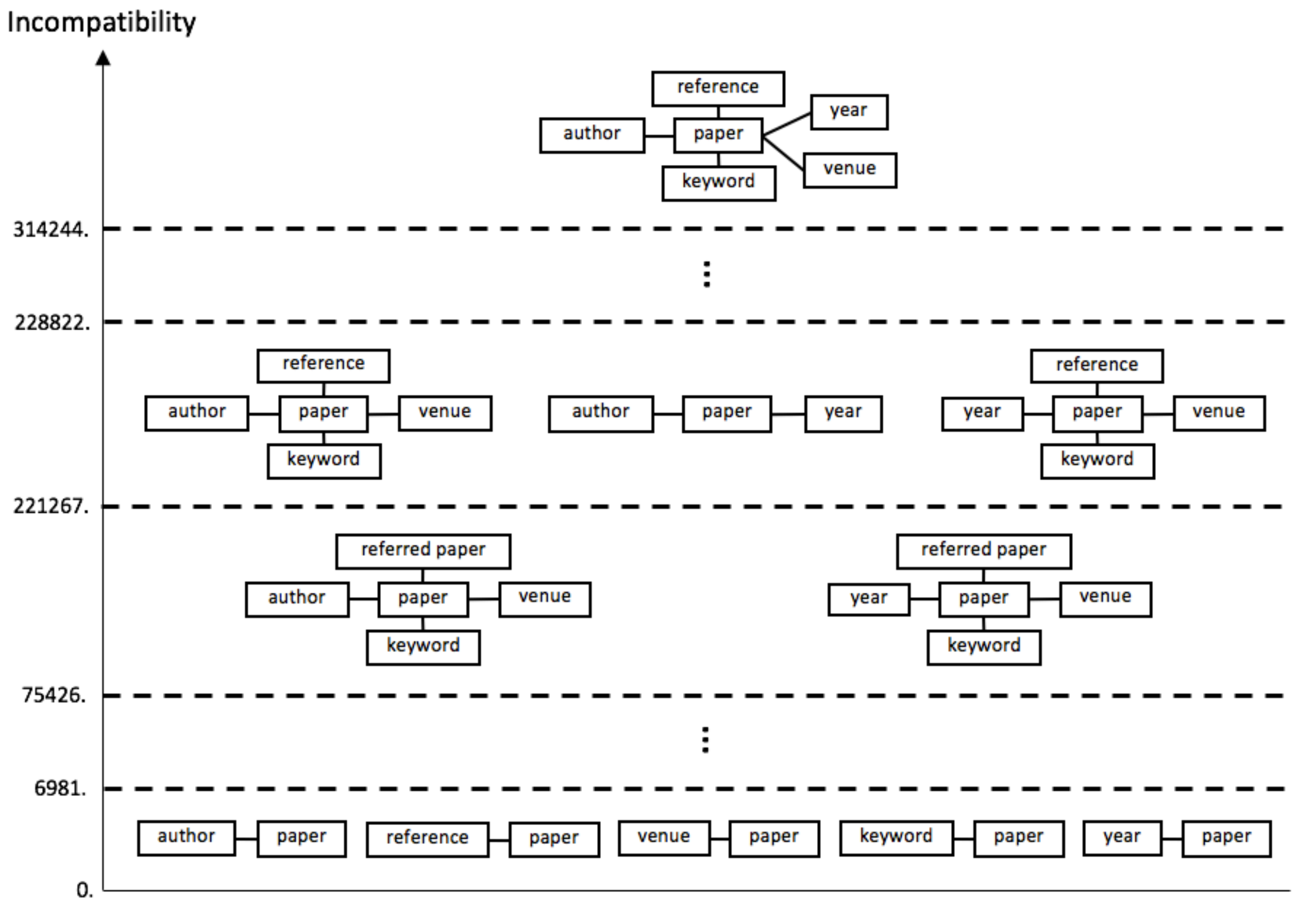
$$\gamma(u) := \frac{\sum_{\phi(\tilde{u})=\phi_c} \max\left\{\mathbf{P}_{u,:}^{\psi_r}(\mathbf{P}_{\tilde{u},:}^{\psi_r})^\top, \mathbf{P}_{u,:}^{\psi_l^{-1}}(\mathbf{P}_{\tilde{u},:}^{\psi_l^{-1}})^\top\right\}}{\sum_{\phi(\tilde{u})=\phi_c} \min\left\{\mathbf{P}_{u,:}^{\psi_r}(\mathbf{P}_{\tilde{u},:}^{\psi_r})^\top, \mathbf{P}_{u,:}^{\psi_l^{-1}}(\mathbf{P}_{\tilde{u},:}^{\psi_l^{-1}})^\top\right\}} - 1$$

- This measure satisfies properties: non-negativity, monotonicity, and convexity w.r.t. aspects.
- We will further validate its effectiveness by experiments.

The AspEm Framework – Aspect Selection

AspEm selects a set of aspects with incompatibility **below a specified threshold.**

More general (bigger), more incompatibility



More specific (smaller), less incompatibility

The AspEm Framework – Embedding One Aspect

To embed each selected representative aspect:

- AspEm is a flexible framework, and one can choose their favorite network embedding algorithm.
- In our instantiation, we adapt the LINE (WWW'15) [1] algorithm and further distinguish edge types.

The probability inferred from embedding:

$$p^a(v|u,r) = \frac{\exp\left(\mathbf{f}_u^a \cdot \mathbf{f}_v^a\right)}{\sum_{v' \in \mathcal{V}:\, \phi(v')=\phi(v)} \exp\left(\mathbf{f}_u^a \cdot \mathbf{f}_{v'}^a\right)}$$

The empirical probability observed in data:

$$\hat{p}^a(v|u,r) = W_{uv}^{(r)} / D_u^{O(r)}$$

Minimizing the difference between the two probabilities is equivalent to minimizing the following objective function:

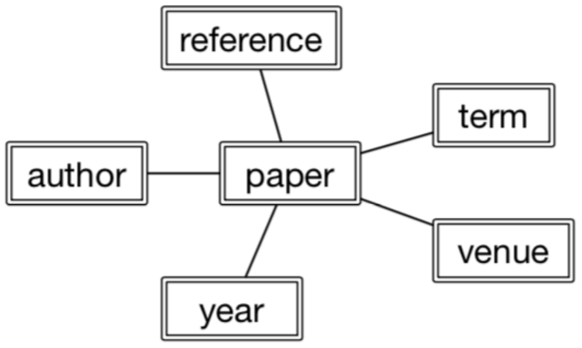
$$\mathcal{O}^a = -\sum_{r \in \mathcal{R}^a} \frac{1}{\Omega^{(r)}} \sum_{u \in \mathcal{V}_{O(r)}} W_{uv}^{(r)} \log p^a(v|u,r)$$

[1] Tang, et al. "Line: Large-scale information network embedding." In WWW, 2015.

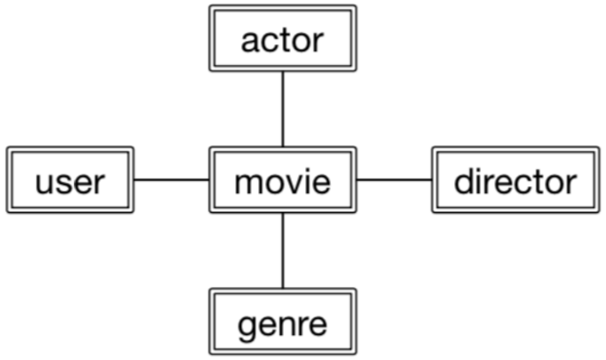
Experiments

Datasets and evaluation tasks

- DBLP: a bibliographical network in the computer science domain:
  - Link prediction task: author identification – to infer the authors of a paper.
  - Classification task: inferring the research group and the research area of authors.
- IMDb: a movie reviewing network.
  - Link prediction task: predicting if a user will review a movie.



(a) DBLP



(b) IMDb

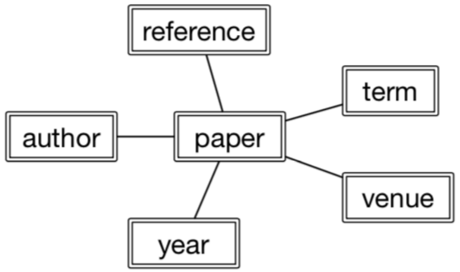
Table 1: Basic statistics for the DBLP and IMDb networks.

DBLP	Author	Paper	Reference	Term	Venue	Year
	1,003,836	1,756,680	693,406	402,687	7,528	62
IMDB	User	Movie	Actor	Director	Genre	
	943	1,360	42,275	918	23	

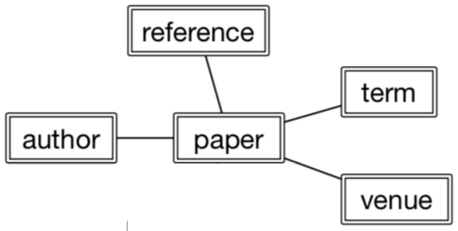
Experiments

Aspects selected by AspEm

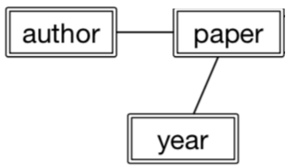
- DBLP: APRTV and APY



(a) DBLP

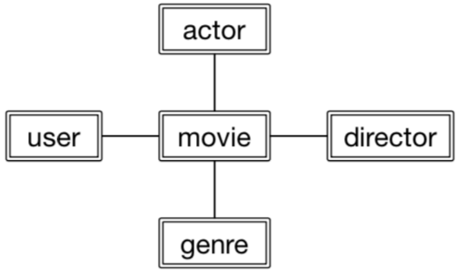


Aspect 1: APRTV

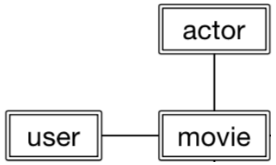


Aspect 2: APY

- IMDb: UMA, UMD and UMG

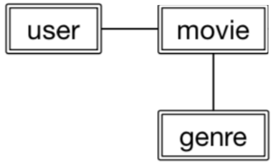


(b) IMDb



Aspect 1: UMA

Aspect 2: UMD



Aspect 3: UMG

Experiments

Baselines

- SVD: a matrix factorization based method.
- DeepWalk [2]: a homogeneous network embedding method, which samples multiple walks starting from each node. Equivalent to node2vec [3] under default parameters.
- LINE [4]: a homogeneous network embedding method, which considers first-order and second-order neighbors.
- OneSpace: a heterogeneous network embedding method and an **ablated version of AspEm**. It uses heterogeneous negative sampling to distinguish node types, but do not model aspects or embed into multiple metric spaces.

[2] Perozzi, et al. "Deepwalk: Online learning of social representations." In KDD, 2014.
[3] Grover, et al. "node2vec: Scalable feature learning for networks." In KDD, 2016.
[4] Tang, et al. "Line: Large-scale information network embedding." In WWW, 2015.

Experiments

Table 3: Link prediction results on DBLP and IMDb.

Dataset	DBLP						IMDb					
Metrics	$P@1$	$P@3$	$P@10$	$R@1$	$R@3$	$R@10$	$P@1$	$P@3$	$P@10$	$R@1$	$R@3$	$R@10$
SVD	0.6648	0.5164	0.2274	0.2939	0.6178	0.8512	0.2470	0.2474	0.2249	0.0152	0.0445	0.1343
DeepWalk	0.7395	0.5297	0.2303	0.3268	0.6329	0.8622	0.3499	0.3605	0.3416	0.0253	0.0774	0.2236
LINE	0.7404	0.5367	0.2299	0.3267	0.6375	0.8596	0.4782	0.4701	0.4130	0.0379	0.1133	0.3137
OneSpace	0.7440	0.5381	0.2279	0.3301	0.6401	0.8519	0.4665	0.4386	0.3852	0.0435	0.1146	0.3038
ASPEM	**0.7724**	**0.5645**	**0.2356**	**0.3479**	**0.6749**	**0.8810**	**0.5090**	**0.4853**	**0.4219**	**0.0464**	**0.1296**	**0.3420**

- AspEm **uniformly outperformed** all four baselines in both link prediction and classification tasks.
- In particular, AspEm yielded better results than OneSpace, which confirms our intuition that **incompatibility can exist among aspects**, and **explicitly modeling aspects** can help better preserve the semantics of an HIN.

Table 2: Classification accuracy in two DBLP tasks.

Dataset/task	DBLP-group		DBLP-area	
Classifier	LR	SVM	LR	SVM
SVD	0.7566	0.7550	0.8158	0.8008
DeepWalk	0.6629	0.7077	0.8308	0.8390
LINE	0.7037	0.7314	0.8526	0.8540
OneSpace	0.7685	0.8333	0.8758	0.8731
ASPEM	**0.8425**	**0.8889**	**0.8786**	**0.8813**

Experiments

Are the representative aspects selected by the incompatibility measure of AspEm really good?

- We exhaust and experiment with **all comparable combinations** of aspects in the DBLP link prediction task.

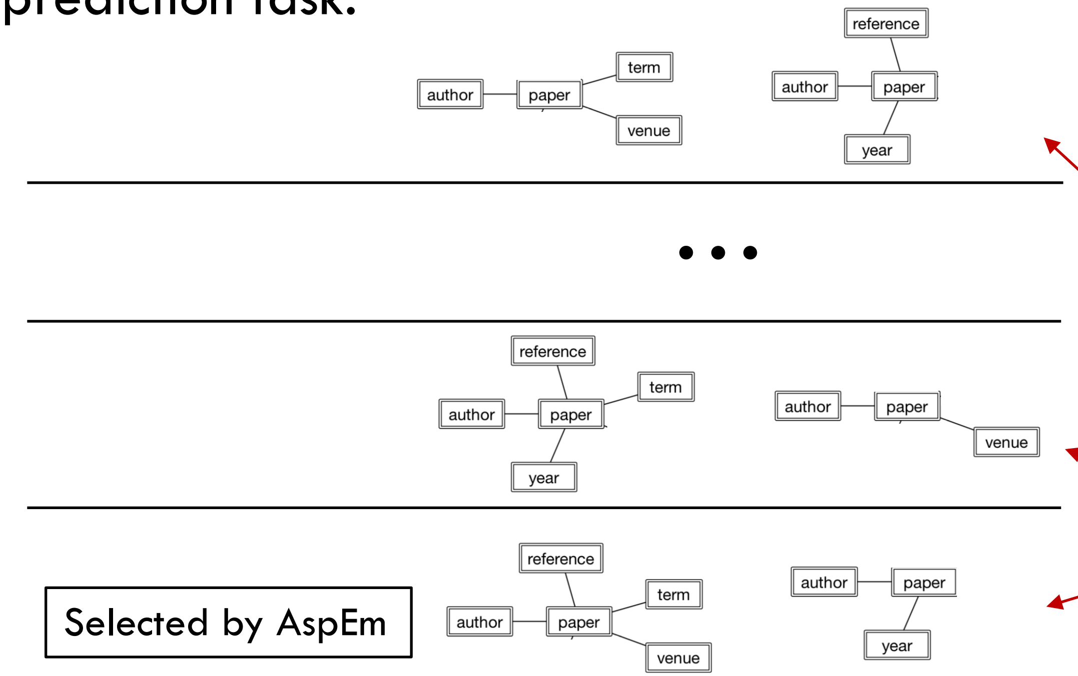


Table 5: Link prediction results using different 2-combinations aspects on DBLP.

Metrics	$P@1$	$P@3$	$P@10$	$R@1$	$R@3$	$R@10$
{APTV, APRY}	0.7522	0.5476	0.2303	0.3362	0.6524	0.8611
{APRV, APTY}	0.7347	0.5327	0.2257	0.3271	0.6327	0.8425
{APRT, APVY}	0.7579	0.5556	0.2332	0.3385	0.6614	0.8708
{APTVY, APR}	0.7384	0.5360	0.2277	0.3280	0.6372	0.8499
{APRVY, APT}	0.7353	0.5356	0.2271	0.3263	0.6355	0.8474
{APRTY, APV}	0.7366	0.5362	0.2277	0.3274	0.6364	0.8492
{APRTV, APY}	**0.7724**	**0.5645**	**0.2356**	**0.3479**	**0.6749**	**0.8810**

- The set selected by AspEm indeed perform the best.

Experiments

Parameter study:

- In the DBLP link prediction task, the performance grows as embedding dimension or number of edge sampled increases at first.
- The change becomes less significant when dimension reaches 100, and number of edges sampled reaches 1,000 million.

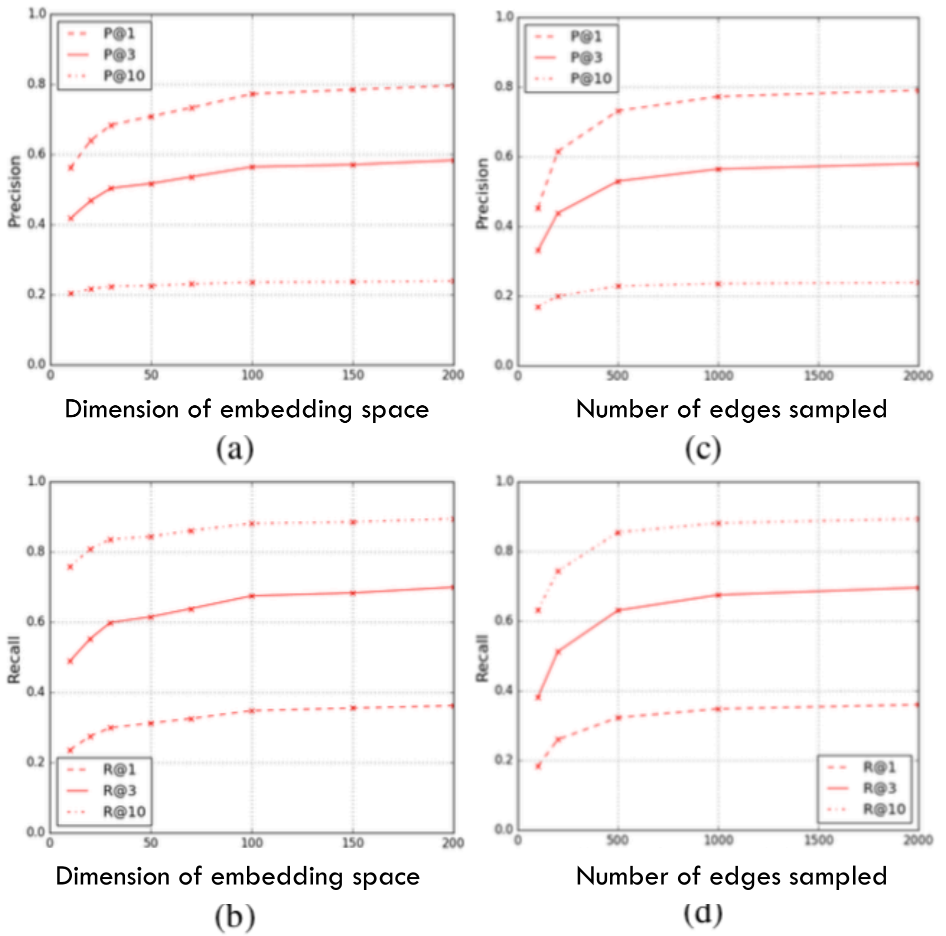


Figure 4: (a) and (b) depict the precision and recall against various dimensions employed for the embedding space. (c) and (d) give the precision and recall against various choices of sampled edge numbers.

Summary

- We provide an insight that **an HIN can have multiple representative aspects that do not align with each other.** We thereby identify that embedding algorithms employing only one metric space may suffer from information loss due to such **incompatibility.**
- We propose **a flexible HIN embedding framework,** named AspEm, that can mitigate the information loss by modeling aspects.
- We propose **a representative aspect selection method** for AspEm **using statistics of HINs without additional supervision.**
- Code available at https://github.com/ysyushi/aspem.