Exploiting Relational Information in Social Networks using Geometric Deep Learning on Hypergraphs

Devanshu Arya
Informatics Institute
University of Amsterdam
d.arya@uva.nl

Marcel Worring
Informatics Institute
University of Amsterdam
m.worring@uva.nl

ABSTRACT

Online social networks are constituted by a diverse set of entities including users, images and posts which makes the task of predicting interdependencies between entities challenging. We need a model that transfers information from a given type of relations between entities to predict other types of relations, irrespective of the type of entity. In order to devise a generic framework, one needs to capture the relational information between entities without any entity dependent information. However, there are two challenges: (a) a social network has an intrinsic community structure. In these communities, some relations are much more complicated than pairwise relations, thus cannot be simply modeled by a graph; (b) there are different types of entities and relations in a social network, taking into account all of them makes it difficult to formulate a model. In this paper, we claim that representing social networks using hypergraphs improves the task of predicting missing information about an entity by capturing higher-order relations. We study the behavior of our method by performing experiments on CLEF dataset consisting of images from Flickr, an online photo sharing social network.

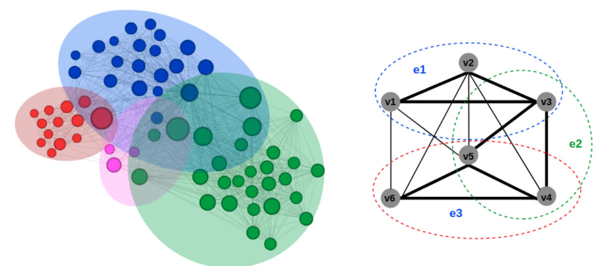
CCS CONCEPTS

• **Computing methodologies** → *Multi-task learning*; *Neural networks*;

KEYWORDS

Social Network; Hypergraph; Geometric Deep Learning

ACM Reference Format:
Devanshu Arya and Marcel Worring. 2018. Exploiting Relational Information in Social Networks using Geometric Deep Learning on Hypergraphs. In *ICMR '18: 2018 International Conference on Multimedia Retrieval, June 11–14, 2018, Yokohama, Japan.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3206025.3206062



(a) Overlapping Community Structure in Social Network

(b) Hypergraph vs Traditional Graph Representation of Relations

Figure 1: Figure (a) represents the overlapping community structure within a social network with the size of node proportional to its degree. As can be seen, the higher degree nodes forms a sub-unit to which other low degree nodes are densely connected. Figure (b) shows a comparison between the hypergraph and traditional graph representation. The higher-order relation between vertices cannot be captured using the pairwise edges. However, it can be easily captured using the hyperedges e1,e2 and e3.

1 INTRODUCTION

The structure of an online social network contains an enormous amount of information within the intrinsic relationships among entities. Capturing implicit relations within these structure allows to perform tasks such as clustering, classification and link prediction. In order to extract these relational information sources, data representation plays a key role. In multimedia, the problem of learning one type of relations between entities to predict other types of relations has been a topic of significant interest. In particular, exploiting relations in online social networks, brings up the problem of generalization across different types of entities. The entities can be users in social communication networks (Facebook, Twitter), images/videos in media sharing networks (Flickr, Instagram), posts in discussion forums (Reddit, Quora) or resources in 'sharing economy' networks (Airbnb, Uber). There exists a multitude of relations within these social networks, even for a small dataset which reveals another hindrance to extract meaningful information. One of the key solutions for these problems is to design a model that can efficiently capture large amounts of relational information between entities, so one can perform tasks irrespective of any entity specific knowledge. Hence, there is a need for a representation which is scalable and a formulation which is neutral to all kinds of social networks.

Traditional graph-based representations of a social network leads to a loss in information, as it implicitly takes into account only pairwise connections between the entities. These pairwise relations fail to represent higher-order relations among the entities. Moreover, a simple binary representation of relations does not depict a collective flow of information. For instance, consider Twitter which

ACM ISBN 978-1-4503-5046-4/18/06…$15.00
https://doi.org/10.1145/3206025.3206062

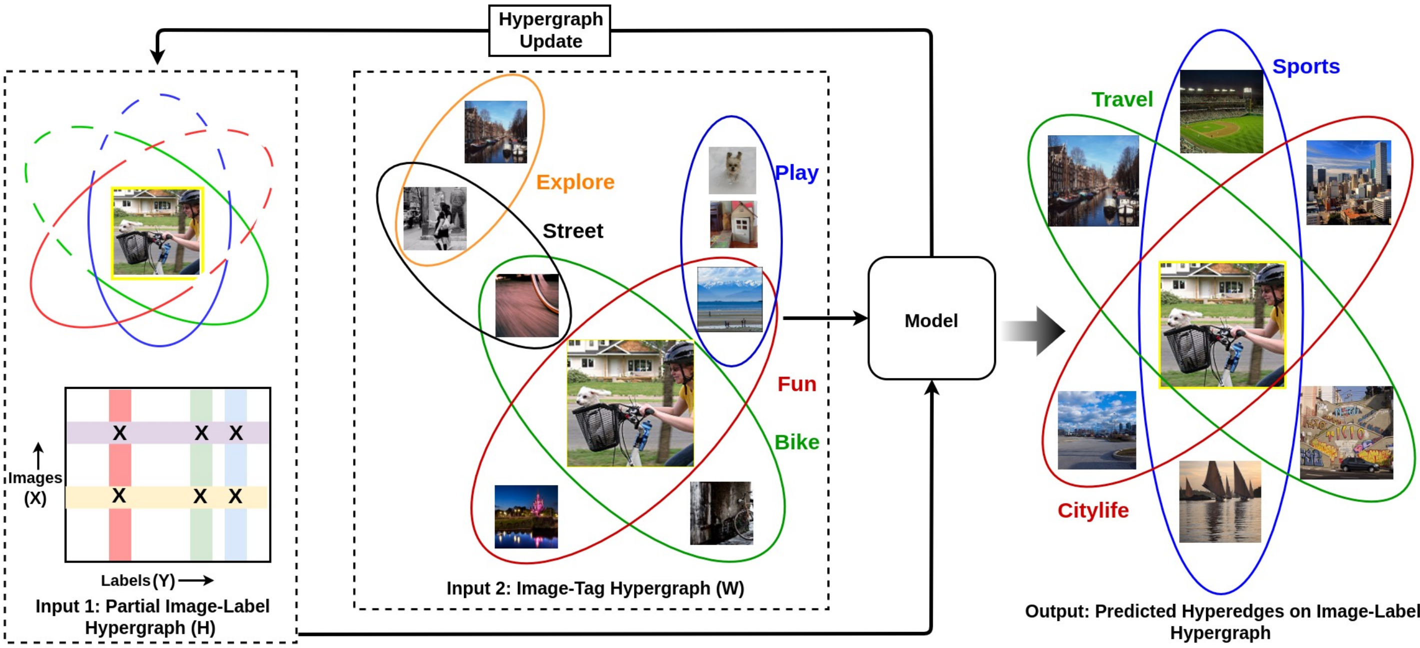


Figure 2: An example of the proposed method on the CLEF dataset of Flickr. The goal is to predict any type of metadata for images given two sets of input. Input 1 is the partial information about the images in one of the metadata spaces represented by an incomplete hypergraph (implying an incomplete incidence matrix). Input 2 is the complete information for the images in the other metadata space. Finally, the output is the set of predicted hyperedges between the images in the partial metadata space.

has users, tweets, hashtags, lists etc.. Representing even a ternary relation of a user releasing a tweet containing multiple hashtags is infeasible using traditional graphs [24]. Or even in the simple case of a co-authorship network, one cannot know whether three or more authors that link together in the network were co-authors of the same paper or not [50] as seen in Fig.1(b). Thus to capture higher-order relations in social networks, traditional graph-based representation of the network proves to be insufficient.

Another characteristic of social networks is the presence of overlapping communities [28]. Most importantly a social network possesses the special property of being a scale-free network [32] [35]. Scale-free networks are a class of power-law networks where the nodes that have many connections (high-degree nodes) tend to be connected to other nodes with many connections, while they are surrounded by many small clusters of low-degree nodes. In other words, social networks contain a structure of communities, where smaller communities in the network are joined to larger communities by highly connected nodes that play the role of local hubs [39]. Graphically, these communities form subunits within the network which show relatively high levels of connection within them and a lower connectivity among them as seen in Fig.1(a). This implies that high-degree nodes in the core of a subunit are crucial for an efficient flow of information and to maintain strong connectivity in these networks. To efficiently capture the relational information within a social network one needs to exploit this dense overlapping community structure.

Various approaches have been proposed in the past to exploit relational information in social networks [31] [10] [36] [19] [29]. However, they do not fully capture the structural features shared within the overlapping community structure of the entities. In order to utilize this property, in this paper we represent social networks as hypergraphs where each entity is represented as a set of vertices and the edges represent the overlapping relations between them. A hypergraph [3] is a generalization of the simple graph in which the edges, called hyperedges, are arbitrary non-empty subsets of the vertex set and may therefore connect any number of vertices. The hyperedges form the key difference between a hypergraph and a traditional graph. The nodes are kept the same as in a simple graph but a hyperedge can connect even all the nodes at once as compared to a traditional graph where an edge is always a connection between 2 nodes. Especially, a set of multimodal entities in a social network can be viewed as a hypergraph whose vertices are the individuals and whose hyperedges are the communities. Finally, a hypergraph representation can be computationally advantageous as compared to the simple graph model since the incidence matrix of a hypergraph requires less storage space in depicting the same volume of information [47]. In this way, a hypergraph is a natural framework to capture the community structure as well as higher-order relations between nodes in the network.

In order to infer relations between entities using only its relational information requires not only a good representation of data, but also a robust powerful model to integrate them. In this work, we propose a methodology which can perform multiple tasks and can be generalized to all social networks. We develop a model to predict missing information (metadata) about an entity by learning relations between entities, without requiring any content-specific features of the entity. To build a multi-functional model for predicting missing metadata, we introduce a multi-graph convolutional neural network model for hypergraphs based on the recent works in deep learning on graphs, specifically graph convolutional networks.

Deep convolutional neural networks [22] have been proven to offer an efficient framework to extract deep meaningful statistical patterns in signals like image, speech or video, in which there is a latent Euclidean structure. However, most of the definitions of convolution, utilize the properties of stationarity and locality which holds for Euclidean data spaces. Recent works [13] [21] on *geometric deep learning* aims to extend the framework of convolutional neural networks to data represented on graphs. The key idea in geometric deep learning is to devise a method for representation learning that can capture structural information within non-Euclidean domains, especially graphs. The applications of graph convolutional network ranges from describing shapes in different human poses [42], semi-supervised classification of authors in citation networks [21] and learning molecular fingerprints [11]. However, regular graph CNNs provide only a partial solution for learning dense information within a group of nodes. The two basic drawbacks of such models in a social network scenario have been due to the compact structure of data which makes the model unscalabale and the inability to share relational information across entities. Hence, we focus on developing a framework that can represent the scale-free properties of a social network and is generalizable to all social networks.

The points below highlight the contributions of this paper:

- We propose a generic framework which can transfer relational information from one type of relation to predict other types of relations between entities. Our approach is entity independent and captures higher-order relations by using hypergraph-based representation of a social network.
- We formulate a model for geometric deep learning on hypergraphs to perform tasks such as multi-label classification, link prediction and recommendation. Our results shows significant improvement as compared to previous graph-based methods.
- We further establish that a hypergraph-based representation of a social network is the most efficient way to build a model for learning the same volume of information in a network as compared to traditional pairwise simple or weighted graphs.

2 BACKGROUND

In this section, we introduce some background on the three concepts on which we base our methodology i.e. hypergraphs, matrix completion and geometric deep learning.

2.1 Notation and Formulation of a Hypergraph

A hypergraph G is formally represented as $H = (V, E)$, where V is a set of vertices and E is a set of hyperedges where each $e \in E$ is a subset of V. The *degree* of a hyperedge e, denoted as $\delta(e)$, is the number of vertices in e. In case of a simple graph, $\delta(e) = 2$ and hence they are known as "2-graph". The diagonal matrices containing the degrees of all vertices (v) and hyperedges (e) are denoted by D_v and D_e respectively. We say that there is a *hyperpath* between vertices v_1 and v_k when there is a sequence of distinct vertices and hyperedges $v_1, e_1, v_2, e_2, ..., e_{k-1}, v_k$ such that $\{v_i, v_{i+1}\} \subseteq e_i$ for $1 \leq i \leq k-1$. One of the key differences of hypergraphs as compared to pairwise simple graphs is its representation using an incidence matrix. It is represented by a $|V| \times |E|$ matrix H with entries $h(v, e) = 1$ if $v \in e$ and 0 otherwise. A simple graph is commonly represented by a square $|V| \times |V|$ matrix A which is known as the adjacency matrix, such that its element a_{ij} is 1 when there is an edge from vertex i to vertex j and 0 when there is no edge. There are many advantages of the incidence matrix (H) over the adjacency matrix (A) to model relational data [47]. The three key advantages are: (i) the incidence matrix of the hypergraph requires less storage space in comparison with the graph adjacency matrix to represent the same volume of information; (ii) hypergraph incidence matrices require fewer operations for matrix-vector multiplication; and (iii) most importantly, the benefits of using the Laplacian of a hypergraph incidence matrix which will be discussed in detail in section 4.2.

2.2 Matrix Completion

Matrix completion is the task of finding the missing values of a partially observed $p \times q$ matrix M. That is, we only observe a sparse set E of observations $M_{i,j} : \forall(i, j) \in E$, with $|E| << pq$. The goal is to estimate the rest of the values $M_{i,j} \notin E$. A particularly popular model is to assume that the values lie in a smaller subspace, resulting in M being a low-rank matrix, which leads to solving a rank minimization problem. Let $\Xi(\bullet)$ be the projection operator selecting only those entries that lie in the set E and let R be the target matrix to be reconstructed using $M_{i,j} : \forall(i, j) \in E$. Then the rank minimization problem is given by:

$$\widetilde{R} = \min_{R \epsilon \mathbb{R}^{p \times q}} rank(R) \text{ s.t. } \Xi(R) = \Xi(M), \tag{1}$$

However, in reality the observed entries always contain a small amount of noise. For example, in movie recommmender systems the rating given by user contains noisy input and in wireless communications the signals received are contaminated with noise. To make the low-rank matrix completion problem more robust to noise, Candes et.al. [8] relaxed eq.1 by nuclear norm minimization given by eq.2.

$$\widetilde{R} = \min_{R} \frac{1}{2} ||R||_* + \lambda_R ||\Xi(M - R)||_F^2 \tag{2}$$

The number of unknown variables in this formulation are in the order of $p \times q$ which makes it practically unscalable for large matrices. One of the solutions is to use a factorized representation of matrix R i.e. $R = XY^T$, where X, Y are $p \times r$ and $q \times r$, matrices respectively with $r << min(p, q)$, which formulates as eq.3 [43].

$$\widetilde{X}, \widetilde{Y} = \min_{X,Y} \frac{1}{2} ||X||_F^2 + \frac{1}{2} ||Y||_F^2 + \lambda_{X,Y} ||\Xi(M - XY^T)||_F^2 \tag{3}$$

Low-rank further implies linear dependence of rows/columns of M which can be utilized to constraint the space of solutions to be smooth. In many scenarios, the rows/columns form communities which can further optimize computation by incorporating proximity information among rows/columns. Recent work on geometric matrix completion has shown the importance of these relations by using them as side information to the matrix completion problem [20] [34] [2] [38]. It assumes that there exists a graph $G_r = (V^X, E^X)$ whose adjacency matrix encodes the relationships between the p rows of X and a graph $G_c = (V^Y, E^Y)$ for q rows of Y. The geometric matrix completion can then be written as

$$\min_{X,Y} \frac{1}{2}||X||^2_{G_r} + \frac{1}{2}||Y||^2_{G_c} + \lambda_{X,Y}||\Xi(M - XY^T)||^2_F \quad (4)$$

where, $||X||^2_{G_r} = trace(X\Delta_r X^T)$ and $||Y||^2_{G_c} = trace(Y\Delta_c Y^T)$ are the graph Dirichlet semi-norm for rows and columns respectively. Δ_r and Δ_c are the row and column laplacian matrices.

2.3 Geometric Deep Learning

As defined by Bronstein et.al. [5]; "Geometric deep learning is an umbrella term for emerging techniques attempting to generalize (structured) deep neural models to non-Euclidean domains such as graphs and manifolds". One of the early attempts on generalizing neural network to graphs are due to Scarselli et.al. in 2005 [15], who proposed a combination of recurrent neural networks and random walk models called Graph Neural Networks (GNN). The first formulation of convolution neural networks on graphs used the definition of convolutions from graph signal processing in the spectral domain [6].

In this work, we focus on applying convolution network on graphs in order to learn the intrinsic relations in social networks. A convolutional layer in the spectral domain is defined as

$$f_l^{out} = \xi(\sum_{l'=1}^{p} \Phi_k \hat{G}_{l,l'} \Phi_K^T f_{l'}^{in}) \quad (5)$$

where, $F_{in}^{n\times p} = (f_1^{in} ... f_p^{in})$ and $F_{out}^{n\times q} = (f_1^{out} ... f_p^{out})$ represent the p and $q-$dimensional input and output signals on the vertices of the graph. Φ_k is the $n \times k$ matrix of the eigenvectors from the spectral decomposition of the graph. $\hat{G}_{l,l'}$ are the learnable spectral filters and ξ is the ReLU non-linearity. Further advancement to this definition have been proposed in order to make a *Graph Convolution Network* (GCN) generic and scalable [16] [11] [21].

3 RELATED WORK

We review two categories of related work: studies on context/network based learning of relational information in social networks and applications of deep learning approaches on graphs.

3.1 Learning Relational Information in Social Networks

Several approaches have been introduced for learning relations within a social network. They can be grouped in five main categories based on their representation of social network data.

In the first one, [26] proposed one of the earliest approaches where they use network topology in which, they model a social network as a simple homogeneous graph where each node represents an entity and each link denotes social relationships. [45] proposed Link Prediction using Social Features (*LPSF*), based on features extracted from patterns of prominent interactions across the network for each entity pair. These features are very useful in identifying similar node pairs, even when they are far away. They propose a simple yet powerful model to capture relations between entities. However, these approaches are not made generic across all networks and lack scalability due to their dependence on pre-defined methods for feature extraction.

In the second type of approach [48] proposed to model social networks as pairwise heterogeneous graphs as opposed to homogeneous ones and apply a random walk algorithm to calculate link proximity. Online photo sharing networks have been of particular interest in learning relations due to the generation of large amounts of metadata. [31] proposed a graphical model that treats image classification as a problem of simultaneously predicting binary labels for a network of photos. They represent each image by a node while the edges are formed between images that have some common property. The first two approaches model social relations between entities as pairs and then apply a structural learning algorithm. These approaches can be scalable to large networks but they still fail to capture any higher-order relations. Therefore, they cannot make use of the community structure in social networks leading to loss in information. Moreover, their applicability to real-world networks is confined as they use parametric methods for modeling relations [19].

The third category of approaches represents data on an ego network, which consist of a focal node ("ego") and the nodes to whom ego is directly connected to ("alters"). Egos and alters are tied to each other by social relations, in [12] and [23], the authors propose to learn social circles by representing the data in ego networks. Li et. al. [25] further study the problem of profiling user attributes in social networks by capturing the correlation between attributes and social connections in an ego network. These approaches however does not generalize for all types of social networks and to learn all kinds of relations between entities and metadata.

Another kind of approaches are based on hypergraph theory. Hypergraph-based models have been widely used in the multimedia domain for solving the problems of community detection [49] [27], multi-label classification [9] [44], tag-based social image searching [14], music recommendation [7] and link prediction in social networks [24]. In this work, we use hypergraph-based approach to represent data, for precisely capturing the high-order relations, in order to build a generic framework for classification, recommendation and link prediction.

Several "non-graph" based approaches to exploit relational information across domains have also been a field of particular interest. Earlier works on multi-domain collaborative filtering includes interaction-associated information of users and items as side information for recommendation. Cross-domain collaborative filtering (CDCF) [18] has recently started to draw significant research attention. The basic concept of CDCF is to borrow rating knowledge for each user from some related auxiliary domains, whose rating matrices are relatively dense, to alleviate the rating sparsity problem in the sparse target domain. These approaches rely mostly on implicit domain correlations that are mined solely from user preference data, and and no explicit links are exploited. There are two major questions surrounding this approach [41]. First, what could be the common knowledge that can be transferred/shared between different domains, and, second, what could be the optimal way to transfer/share knowledge between different domains [37].

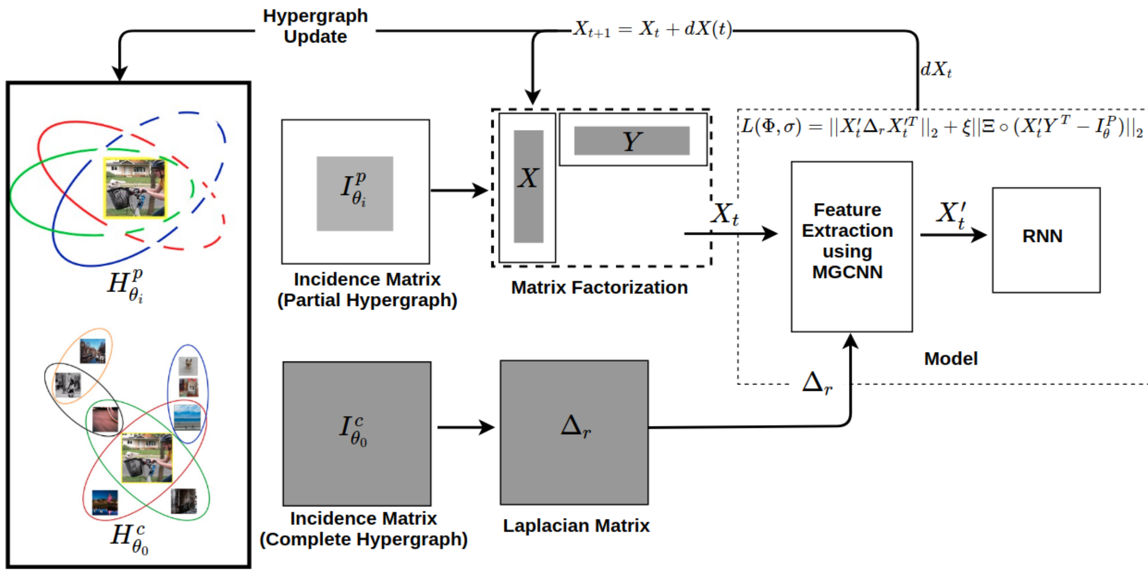


Figure 3: A block-diagram of the proposed model. The input to the model are the partial $H\theta_i^p$ and complete $H\theta_0^c$ hypergraphs corresponding to the two types of metadata θ_i and θ_0 respectively. $I\theta_0^c$ and $I\theta_i^p$ are the incidence matrices (the grey color depicts data used for training whereas white represents missing data). The model updates the hypergraph incrementally by updating the incidence matrix using geometric deep learning based model.

3.2 Application of Deep Learning based Approaches on Graphs

There has been a recent surge of interest to formulate deep learning methods on non-euclidean domain especially in graphs. The effectiveness of deep learning graph-based approaches ranges from computer graphics [4] to chemistry [13]. The spectral graph convolutional neural networks (*GCN*), originally proposed in [6] and extended in [11] have proven effective in classification of handwritten digits and news texts. [21] proposed a simplified *GCN* for semi-supervised classification of authors in a citation network. In the computer vision community, *GCN* has been extended by [30] to describe shapes in different human poses, [42] to demonstrate classification of point clouds and [33] for image and 3D shape analysis. In multimedia, [40] proposed an approach to categorize user posts for political extremism content based on their discussion topics. Deep learning on graphs for social networks is yet to be explored for their ability to uncover hidden relations between multimedia items. In this paper, we take a step further to devise a generic model for learning relations in social networks using geometric deep learning methods.

4 PROPOSED MODEL

In this work, we define a trainable graphical model that treats predicting metadata for an entity, as the unified problem of generating sets of hyperedges across entities. The basic hypothesis of the model is that entities related through one set of metadata carry imperative information which can be learnt to predict other relational properties between them. In this paper, we will use $H^{p/c}_\Theta$ (with $I^{p/c}_\Theta$ as its incidence matrix) to denote a partial(p) or a complete (c) hypergraph. The subscript Θ is the type of metadata used to construct the hyperedges i.e. $\Theta = t/l/g/u$ for tags (t), labels (l), groups (g) and users (u) respectively. The inputs to the model are: (a) a complete hypergraph of entities constructed using one type of metadata denoted by $H^c_\Theta$and (b) a partial hypergraph on the same sets of entities constructed from the required metadata denoted by H^p_Θ. The training of the model has three phases: constructing the model by formulating it as a factorized matrix completion problem, relational feature extraction using geometric deep learning and finally updating the partial hypergraph by predicting hyperedges across entities. Fig.3 shows these three phases as a block diagram.

4.1 Formulating Hyperedge Prediction as Matrix Completion

The computational advantage of using a hypergraph for the above mentioned problems instead of a simple graph is the representation of its vertices and edges by an incidence matrix. As compared to traditional graphs where the incidence matrix has an additional constraint of only two non-zero values in each column ("2-graph" property), the incidence matrix of a hypergraph can have as many as all non-zero values in each column. Therefore, generating hyperedges in a hypergraph can be termed equivalent to the problem of filling missing entries in its corresponding incidence matrix. In this work, we represent the relation between entities using their metadata by hypergraphs. Each entity corresponds to a vertex and the edges depict all unique values of the corresponding metadata. The respective incidence matrix ($I^{p/c}_\Theta$) is of dimension $n \times \theta$, where n is the total number of entities and θ are the unique values corresponding to the metadata Θ. Hence, the problem of predicting hyperedges between entities in H^p_Θ reduces to completing the incidence matrix I^p_Θ with multiple missing entries corresponding to each column and at least one known entry in each row, where each row is an entity and the columns are values of the metadata.

In order to incorporate relational information from H^c_Θ to fill I^p_Θ, we further take motivation from geometric matrix completion. This is achieved by factorizing I^p_Θ into its row (X_p) and column (Y_p) matrices such that $I^p_\Theta = X_p Y_p^T$. The matrix X with dimension $n \times q$ represents the entities (vertices) and Y with dimension $\theta \times q$ represents values of metadata (edges), with $q << min(n, \theta)$. The implicit relation between the individual entities in X_p (row matrix) is captured using H^c_Θ. We consider identity relations between values in the column-matrix Y_p i.e. each value of the metadata is related to itself only. This is because the unique values of metadata (Θ) do not contain any relational information among them. That is, we treat all the values of a metadata to be independent and uniformly distributed in space. Thus eq.4 can be re-formulated with $H^c_\Theta \equiv G_r$, I^p_Θ as the partially filled matrix M and $||Y||_{Gc}$ as a constant, which leads to the following minimization problem

$$\min_X ||X||^2_{H^c_\Theta} + \lambda_X ||\Xi \circ (I^p_\Theta - XY^T)||^2_F \tag{6}$$

4.2 Feature Extraction using Multi-Graph CNNs on Hypergraph

The second phase of our model aims at jointly extracting features from H^c_Θ and H^p_Θ. In this way, we can transfer the relational information from the complete hypergraph H^c_Θ to predict the missing hyperedges in H^p_Θ. In this paper, we devise our solution based on recent work on multi-graph convolution ($MGCNN$) [34]. It uses the formulation for GCN using recurrent Chebyshev polynomials which simplifies eq.5 [11]. The motivation behind multi-graph convolution is that, a Fourier transform of a 2-dimensional signal can be simplified by formulating it as applying a one-dimensional Fourier transform to its rows and columns. In particular, multi-graph convolution proposes a method of matrix completion, given the rows and columns of a matrix possess relational information within themselves. In our framework, we extract features combining I^p_Θ and I^c_Θ by stacking multi-graph CNN layers given by

$$X'_t = \sum_{j=0}^{q} \Phi_j T_j(\Delta_r) X_t \tag{7}$$

where Φ_j are the learnable filter coefficients, $\Delta_r^{n\times n}$ is the row-hypergraph Laplacian and T_j is the representation of filters using Chebyshev polynomials. In this way a multi-graph CNN on $X_t^{n\times q}$ with a single channel produces a k dimensional output $X_t'^{n\times q\times k}$.

The other advantage of the above formulation is the use of the Laplacian to encode information from data defined on hypergraph H^c_Θ. The Laplacian matrix of a hypergraph has been shown to be useful for learning higher-order relations [1] [44], spectral clustering of edges [50] and to measure the relatedness between two entities [7]. In this paper, we use the normalized hypergraph Laplacian matrix (Δ_r) [50] given by

$$\Delta_r = \mathbb{I} - D_v^{-\frac{1}{2}} I^c_\Theta D_e^{-1} I^{cT}_\Theta D_v^{-\frac{1}{2}} \tag{8}$$

where D_v and D_e are the vertex and edge degree matrices of hypergraph H^c_Θ respectively, $\mathbb{I}$ is the identity matrix and I^{cT}_Θ is the transpose of incidence matrix I^c_Θ. The Laplacian will be used for incorporating the structure of the hypergraph H^c_Θ in eq.7. In this way, we extract relational features using combined information from the complete and the partial hypergraph.

	Task1	Task2	Task3	Task4
θ_0	Tags (t)	Tags(t)	Tags (t)	Labels (l)
θ_i	Labels (l)	User (u)	Groups (g)	Tags (t)
$\|\|rel_{\theta_i}\|\|$	613,014	51,804	70,226,414	91,485,864
$\|\|rel(H^c_{\theta_0})\|\|$	45,766	45,766	45,766	55,396
$\|\|rel(G^c_{\theta_0})\|\|$	85,802	85,802	85,802	95,766

Table 1: Table showing the details about the 4 tasks. The goal is to predict relations given a partial set of rel_{θ_i} and complete set of relations represented on hypergraph ($rel(H^c_{\theta_0})$) or on simple and weighted-graph ($rel(G^c_{\theta_0})$).

4.3 Incremental Updates of the Hypergraph

The next step is to diffuse the features extracted by coupling the structures of the two hypergraphs (H^c_Θ and H^p_Θ). The partial hypergraph is updated incrementally as a consequence of the completion of its incidence matrix. We use a Recurrent Neural Network (RNN) [17] to predict small incremental changes (dX) to the matrix X [34]. One of the main advantages of using an RNN for predicting accurate small changes is its ability to store information for longer temporal steps. The model is finally trained by feeding the features extracted from multi-graph CNN (X'_t) to an RNN and perform training by using the minimization eq.6 in geometric matrix completion as the loss function.

$$L(\Phi, \sigma) = ||X'_{t,\sigma} \Delta_r X'^{\ T}_{t,\sigma}||_2 + \lambda_X ||\Xi \circ (X'_{t,\sigma} Y^T - I^P_\theta)||_2 \tag{9}$$

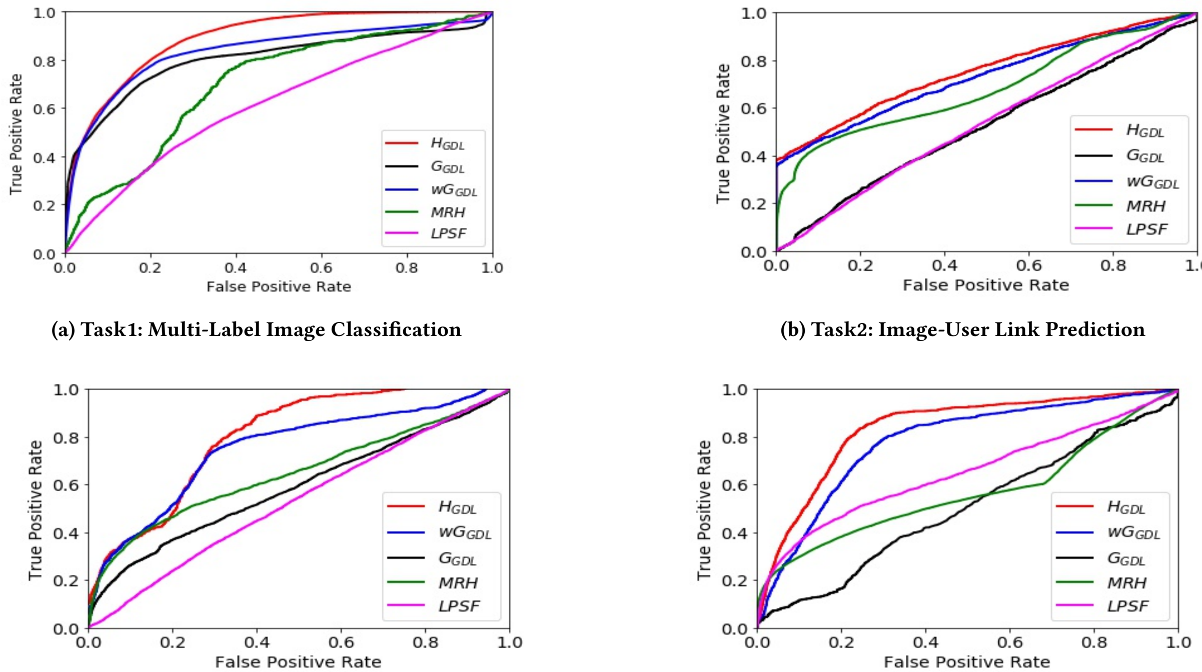
where $X't$ is the feature extracted by multi-graph convolution with Φ as the learning coefficient, σ denotes the parameter for RNN and the subscript t denotes the number of diffusion iterations.

5 EXPERIMENTS

In this section, we perform extensive experiments to show the advantages of learning higher-order relations in a social network using geometric deep learning on hypergraphs as compared to other approaches. We design our experiments to investigate the following:

- Performance of the proposed generic framework to predict multiple types of relations between entities
- Advantages of using geometric deep learning over existing simple graph as well as hypergraph-based learning
- Efficiency in representing relational information of a network using hypergraphs as compared to pairwise simple graph representation

To evaluate our model, we explore the online photo sharing social network Flickr, which generates a huge amount of metadata and hence relations for each image. Flickr has been particularly very popular in using social network metadata for image classification among other implications [19] [31]. The metadata, such as user-generated tags and community-curated groups in Flickr are used by people as a means to communicate with other people, and



(a) Task1: Multi-Label Image Classification

(b) Task2: Image-User Link Prediction

(c) Task3: Group Recommendation

(d) Task4: Tag Recommendation

Figure 4: Experiment 1 - Receiver Operating Characteristics (ROC) curve showing the performance of the models on each of the 4 tasks. The hypergraph-based geometric deep learning model (H_{GDL}) has significant advantage as compared to other methods.

as a means to describe the image and its location. But not every image is annotated with all the information, hence using relational information can be highly informative in unveiling the missing information of every image.

Data Setup For our experiments, we study the CLEF dataset [31] comprising of images from Flickr which has social network metadata and has labels provided by human annotators for each image. The dataset consists of 4,546 images with 99 labels (l), 21,192 tags (t), 10,575 groups (g) and 2,663 users (u). We show that our framework can be used as a generic multi-functional setup for generating information for an image by performing 4 types of tasks using our model: $Task1$: Multi-Label Image Classification, $Task2$: Image-User Link Prediction, $Task3$: Group Recommendation and $Task4$: Tag Recommendation. Given a set of known-metadata (θ_0) for each image, we first construct the complete hypergraph $H^c_{\theta_0}$. Our goal is to predict other sets of partially known metadata (θ_i) associated with the images.

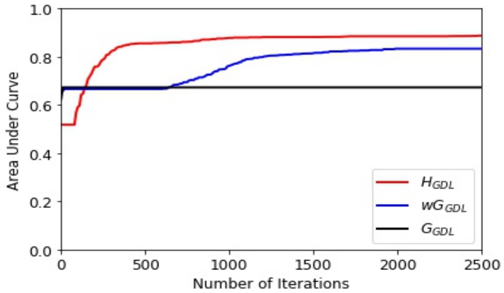
Training The total number of relations, $||rel_{\theta_i}||$ between the images and the target metadata (θ_i) is tabulated in Table1. As seen from the table, each image has multiple values of metadata in common with other images, resulting in a multitude of relations. We randomly sample 40% of these relations and keep them aside to use them as test set. The remaining relations are used to construct the partial hypergraph $H^p_{\theta_i}$ for training the model along with the complete hypergraph $H^c_{\theta_0}$.

Evaluation To show the efficiency in representing social network information with a hypergraph, we compare our result with the data represented by hypergraph (H), simple graph (G) and weighted graph (wG) using the same model. Simple graph (G) indicates a binary relation between entities with a value 1 if the two entities share at least one common value of the metadata. The weighted graph (wG) is constructed by assigning weights equal to the count of values of the metadata common between two entities. The hypergraph-based representation reduces the total number of relations between entities and the known metadata significantly by representing higher-order relations as community. This can be seen from Table1 where $||rel(H^c_{\theta_0})||$ and $||rel(G^c_{\theta_0})||$ denote the total number of relations in a hypergraph and graph based representation respectively.

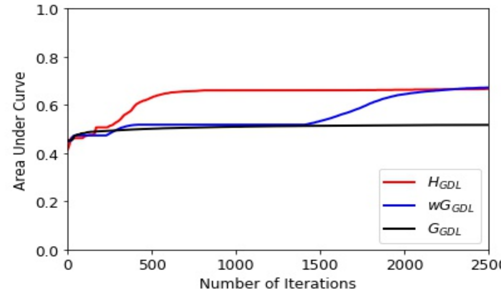
We evaluate the performance of our geometric deep learning based model as compared to the previous hypergraph based algorithm (*MRH*) [7] [24] and a graph-based model trained on social network features (*LPSF*) [45] [46] for the same tasks. *LPSF* as mentioned under the first approach in section 3.1, trains a neural network on popular features like Page Rank, Number of Common Neighbors, Preferential Attachment etc. extracted from a social network. We use the notation H_{GDL}, wG_{GDL} and G_{GDL} for our geometric deep learning (*GDL*) model on hypergraph, weighted graph and regular graph respectively.

5.1 Experiment 1: Learning Relational Information in a Social Network

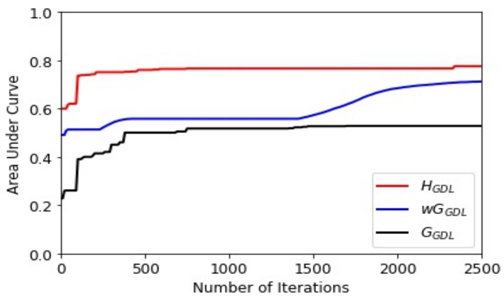
We start our experimental evaluation by showing the performance of our model, *MRH* and *LPSF* on the 4 tasks. To evaluate the performance of our model and show its advantages over other methods, we show the Receiver Operating Characteristic (ROC) curves for each tasks. The ROC curve depicts how well a model is able to



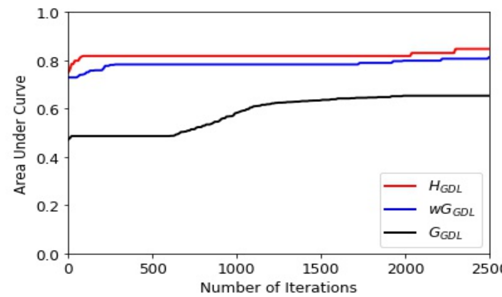
(a) Task1: Multi-Label Image Classification



(b) Task2: Image-User Link Prediction



(c) Task3: Group Recommendation



(d) Task4: Tag Recommendation

Figure 5: Experiment 2 - Figure showing the rate of learning with each iteration of the proposed model using hypergraph (H_{GDL}), weighted graph (wG) and simple graph (G). As can be seen, the hypergraph-based model converges faster for all the 4 tasks implying a better representation to learn relational information.

predict the presence/absence of a relation among images with the corresponding metadata. Fig.4 shows the performance of the models on the 4 tasks. The Geometric Deep Learning based approach outperforms existing hypergraph-based *MRH* and graph-based *LPSH* methods in all the 4 tasks. This confirms, the significant advantage of using a hypergraph representation of the network as compared to simple and weighted graphs using the same model. Most importantly, this proves the advantage of learning relations using geometric deep learning techniques as compared to existing hypergraph-based model.

5.2 Experiment 2: Measuring the efficiency of representing data from social network

To explore the advantage of representing a social network using hypergraphs as compared to traditional graphs, we evaluate their efficiency in learning relational information. We compare the rate of convergence of our algorithm on the three graph frameworks mentioned above i.e. hypergraph (H), weighted graph (wG) and simple graph (G) on the 4 tasks. The faster the algorithm converges, the better the framework is in capturing the same volume of relational information. We plot the area under the ROC curve against the number of iterations used to update the matrix incrementally. As can be seen from Fig.5, the hypergraph-based representation converges faster than simple and weighted graphs for all the 4 tasks. This concludes the efficiency of a hypergraph in capturing information which makes it the best choice to represent data on a social network.

6 CONCLUSIONS

In this paper, a generic method to exploit relational information between entities in a social network for predicting missing information about an entity has been presented. In contrast with traditional graph representation, we model a social network using hypergraphs. We show the importance of using hypergraphs in order to capture all types of entities and either the pair wise or high-order relations among them to avoid loss of any information. Moreover, our approach is content independent i.e. it does not depend on any entity-specific information and hence can be generalized to all types of social networks. We formulate the learning problem as matrix completion on graphs and extend the methods on geometric deep learning to hypergraphs. We evaluate our model on 4 tasks: multi-label image classification, image-user link prediction, group and tag recommendation in a Flickr dataset. Experimental results show a significant advantage in representing social networks by hypergraphs and using deep learning based method for exploiting relational information within the network. We also prove the computational effectiveness of representing the same volume of information from a social network on a hypergraph as compared to the traditional pairwise graphs.

ACKNOWLEDGMENTS

This research has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 700381.

REFERENCES

[1] Sameer Agarwal, Kristin Branson, and Serge Belongie. 2006. Higher order learning with graphs. In *Proceedings of the 23rd international conference on Machine learning*. ACM, 17–24.

[2] Kirell Benzi, Vassiiis Kalofolias, Xavier Bresson, and Pierre Vandergheynst. 2016. Song recommendation with non-negative matrix factorization and graph total variation. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. Ieee, 2439–2443.

[3] Claude Berge. 1973. *Graphs and Hypergraphs (North-Holland mathematical library; v. 6)*. Elsevier.

[4] Davide Boscaini, Jonathan Masci, Emanuele Rodolà, and Michael Bronstein. 2016. Learning shape correspondence with anisotropic convolutional neural networks. In *Advances in Neural Information Processing Systems*. 3189–3197.

[5] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. 2017. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine* 34, 4 (2017), 18–42.

[6] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. 2014. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*.

[7] Jiajun Bu, Shulong Tan, Chun Chen, Can Wang, Hao Wu, Lijun Zhang, and Xiaofei He. 2010. Music recommendation by unified hypergraph: combining social media information and music content. In *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 391–400.

[8] Emmanuel J Candes and Yaniv Plan. 2010. Matrix completion with noise. *Proc. IEEE* 98, 6 (2010), 925–936.

[9] Gang Chen, Jianwen Zhang, Fei Wang, Changshui Zhang, and Yuli Gao. 2009. Efficient multi-label classification with hypergraph regularization. In *Proceedings of Computer Vision and Pattern Recognition, 2009. CVPR*. IEEE, 1658–1665.

[10] Jinpeng Chen, Hongbo Gao, Zhenyu Wu, and Deyi Li. 2013. Tag co-occurrence relationship prediction in heterogeneous information networks. In *Parallel and Distributed Systems (ICPADS), 2013 International Conference on*. IEEE, 528–533.

[11] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*. 3844–3852.

[12] Christopher P Diehl, Galileo Namata, and Lise Getoor. 2007. Relationship identification for social network discovery. In *AAAI*, Vol. 22. 546–552.

[13] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. 2015. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in neural information processing systems*. 2224–2232.

[14] Yue Gao, Meng Wang, Huanbo Luan, Jialie Shen, Shuicheng Yan, and Dacheng Tao. 2011. Tag-based social image search with visual-text joint hypergraph learning. In *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 1517–1520.

[15] Marco Gori, Gabriele Monfardini, and Franco Scarselli. 2005. A new model for learning in graph domains. In *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, Vol. 2. IEEE, 729–734.

[16] Mikael Henaff, Joan Bruna, and Yann LeCun. 2015. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163* (2015).

[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[18] Liang Hu, Jian Cao, Guandong Xu, Jie Wang, Zhiping Gu, and Longbing Cao. 2013. Cross-domain collaborative filtering via bilinear multilevel analysis. In *International Joint Conference on Artificial Intelligence*. IJCAI/AAAI.

[19] Justin Johnson, Lamberto Ballan, and Li Fei-Fei. 2015. Love thy neighbors: Image annotation by exploiting image metadata. In *Proceedings of the IEEE international conference on computer vision*. 4624–4632.

[20] Vassilis Kalofolias, Xavier Bresson, Michael Bronstein, and Pierre Vandergheynst. 2014. Matrix completion on graphs. *arXiv preprint arXiv:1408.1717* (2014).

[21] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[23] Jure Leskovec and Julian J Mcauley. 2012. Learning to discover social circles in ego networks. In *Advances in neural information processing systems*. 539–547.

[24] Dong Li, Zhiming Xu, Sheng Li, and Xin Sun. 2013. Link prediction in social networks based on hypergraph. In *Proceedings of the 22nd International Conference on World Wide Web*. ACM, 41–42.

[25] Rui Li, Chi Wang, and Kevin Chen-Chuan Chang. 2014. User profiling in an ego network: co-profiling attributes and relationships. In *Proceedings of the 23rd international conference on World wide web*. ACM, 819–830.

[26] David Liben-Nowell and Jon Kleinberg. 2007. The link-prediction problem for social networks. *journal of the Association for Information Science and Technology* 58, 7 (2007), 1019–1031.

[27] Yu-Ru Lin, Jimeng Sun, Paul Castro, Ravi Konuru, Hari Sundaram, and Aisling Kelliher. 2009. Metafac: community discovery via relational hypergraph factorization. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 527–536.

[28] Dajie Liu, Norbert Blenn, and Piet Van Mieghem. 2012. A social network model exhibiting tunable overlapping community structure. *Procedia Computer Science* 9 (2012), 1400–1409.

[29] Yun-Fu Liu, Jing-Ming Guo, and Lingling An. 2017. Multimedia Classification Using Bipolar Relation Graphs. *IEEE Transactions on Multimedia* 19, 8 (2017), 1860–1869.

[30] Jonathan Masci, Davide Boscaini, Michael Bronstein, and Pierre Vandergheynst. 2015. Geodesic convolutional neural networks on riemannian manifolds. In *Proceedings of the IEEE international conference on computer vision workshops*. 37–45.

[31] Julian McAuley and Jure Leskovec. 2012. Image labeling on a network: using social-network metadata for image classification. *Computer Vision–ECCV 2012* (2012), 828–841.

[32] Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. 2007. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 29–42.

[33] Federico Monti, Davide Boscaini, and Jonathan Masci. 2017. Geometric deep learning on graphs and manifolds using mixture model CNNs. In *Proceedings of Computer Vision and Pattern Recognition, CVPR*.

[34] Federico Monti, Michael Bronstein, and Xavier Bresson. 2017. Geometric matrix completion with recurrent multi-graph neural networks. In *Advances in Neural Information Processing Systems*. 3700–3710.

[35] André A Moreira, Demétrius R Paula, Raimundo N Costa Filho, and José S Andrade Jr. 2006. Competitive cluster growth in complex networks. *Physical Review E* 73, 6 (2006), 065101.

[36] Zhenxing Niu, Gang Hua, Xinbo Gao, and Qi Tian. 2014. Semi-supervised relational topic model for weakly annotated image recognition in social media. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4233–4240.

[37] Weike Pan, Nathan N Liu, Evan W Xiang, and Qiang Yang. 2011. Transfer learning to predict missing ratings via heterogeneous user feedbacks. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, Vol. 22. 2318.

[38] Nikhil Rao, Hsiang-Fu Yu, Pradeep K Ravikumar, and Inderjit S Dhillon. 2015. Collaborative filtering with graph information: Consistency and scalable methods. In *Advances in neural information processing systems*. 2107–2115.

[39] Erzsébet Ravasz and Albert-László Barabási. 2003. Hierarchical organization in complex networks. *Physical Review E* 67, 2 (2003), 026112.

[40] Stevan Rudinac, Iva Gornishka, and Marcel Worring. 2017. Multimodal Classification of Violent Online Political Extremism Content with Graph Convolutional Networks. In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*. ACM, 245–252.

[41] Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)* 47, 1 (2014), 3.

[42] Martin Simonovsky and Nikos Komodakis. 2017. Dynamic Edge-Conditioned Filters in Convolutional Neural Networks on Graphs. In *Proceedings of Computer Vision and Pattern Recognition, 2017. CVPR*.

[43] Nathan Srebro, Jason Rennie, and Tommi S Jaakkola. 2005. Maximum-margin matrix factorization. In *Advances in neural information processing systems*. 1329–1336.

[44] Liang Sun, Shuiwang Ji, and Jieping Ye. 2008. Hypergraph spectral learning for multi-label classification. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 668–676.

[45] Xi Wang and Gita Sukthankar. 2013. Link prediction in multi-relational collaboration networks. In *Proceedings of the 2013 IEEE/ACM international conference on advances in social networks analysis and mining*. ACM, 1445–1447.

[46] Xi Wang and Gita Sukthankar. 2014. Link prediction in heterogeneous collaboration networks. In *Social network analysis-community detection and evolution*. Springer, 165–192.

[47] Michael M Wolf, Alicia M Klinvex, and Daniel M Dunlavy. 2016. Advantages to modeling relational data using hypergraphs versus graphs. In *High Performance Extreme Computing Conference (HPEC), 2016 IEEE*. IEEE, 1–7.

[48] Zhijun Yin, Manish Gupta, Tim Weninger, and Jiawei Han. 2010. Linkrec: a unified framework for link recommendation with user attributes and graph structure. In *Proceedings of the 19th international conference on World wide web*. ACM, 1211–1212.

[49] Yi-Liang Zhao, Qiang Chen, Shuicheng Yan, Tat-Seng Chua, and Daqing Zhang. 2013. Detecting profilable and overlapping communities with user-generated multimedia contents in LBSNs. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 10, 1 (2013), 3.

[50] Denny Zhou, Jiayuan Huang, and Bernhard Schölkopf. 2007. Learning with hypergraphs: Clustering, classification, and embedding. In *Advances in neural information processing systems*. 1601–1608.