OverCite: Finding Overlapping Communities in Citation Network

Tanmoy Chakraborty *, Abhijnan Chakraborty †
* Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur, India – 721302
its_tanmoy@cse.iitkgp.ernet.in
† Microsoft Research India
Bangalore, India – 560001
t-abcha@microsoft.com

Abstract—**Citation analysis is a popular area of research, which has been usually used to rank the authors and the publication venues of research papers. With huge number of publications every year, it has become difficult for the users to find relevant publication materials. One simple solution to this problem is to detect communities from the citation network and recommend papers based on the common membership in communities. But, in today's research scenario, many researchers' fields of interest spread into multiple research directions resulting in an increasing number of interdisciplinary publications. Therefore, it is necessary to detect overlapping communities for relevant recommendation.**

In this paper, we represent publication information as a tripartite 'Publication Hypergraph' consisting of authors, papers and publication venues (conferences/journals) in three partitions. We then propose an algorithm called 'OverCite', which can detect overlapping communities of authors, papers and venues simultaneously using the publication hypergraph and the citation network information. We compare OverCite with two existing overlapping community detection algorithms, Clique Percolation Method (CPM) and iLCD, applied on citation network. The experiments on a large real-world citation dataset show that OverCite outperforms other two algorithms. We also present a simple paper search and recommendation system. Based on the relevance judgements of the users, we further prove the effectiveness of OverCite over other two algorithms.

Keywords—***Overlapping communities, citation network, publication hypergraph, hypergraph clustering, recommendation system***

I. Introduction

Many large scale dynamic complex networks can be described through the intricate web of connections among their units. There has been a demanding question to the researchers about the interpretation of the global organization of such networks as the coexistence of their *a priori* highly interconnected structural sub-units (i.e. *communities*) [1]. Majority of the existing algorithms find exclusive communities from large scale networks; while in reality, the actual networks are made of highly overlapping cohesive groups of nodes. For example, in large scale social networks like Facebook[1] or Google Plus[2], users are part of multiple communities including their family members, friends, colleagues etc. Therefore, it is important to discover overlapping communities from large scale networks.

Citation analysis is quite old but still relevant and popular area of research among experts from different domains. Specially the advent of *Automated Citation Indexing* [2] has changed the nature of citation analysis research. Citation indexes can be analysed to determine the popularity and impact of specific articles, authors and journals/conferences[3], which in turn can rank the authors on h-index [3] and venues on impact factor [4]. Citation analysis can also be used to quantitatively assess the relationships between authors from different institutions and schools of thought, and can bring out some interesting insights into the sociology of academia. For example, the empirical work by Guimera et al. [5] has shown that new collaborations between experienced authors are more likely to result in a publication in a high impact journal than the collaborations between novice authors or repeat collaborations between the same two authors.

The rate of growth in scientific publication has been exponential over the years. Odlyzko [6] showed that the number of scientific papers published annually has been doubling every $10-15$ years for the last two centuries. As more and more papers are getting published, it has become very difficult for users to search interesting and related papers, authors and publication venues on their own. Moreover, in today's research scenario, many researchers' fields of interest spread into multiple research directions resulting in an increasing number of interdisciplinary publications. Major venues also extend their related topics into multiple fields. One simple recommendation scheme would be to efficiently detect overlapping communities of papers, authors and venues and recommend papers, authors or venues to the users depending on the common membership in communities detected from these networks. We describe one such recommendation system in Section IV.

Traditionally, a citation network is represented as a simple graph $G = (X, Y)$, where each node $x_i \in X$ represents a paper and a directed edge $y_{ij} \in Y$ pointing from x_i to x_j indicates that the paper corresponding to x_i cites the paper corresponding to x_j as reference. It is usually stored as a list of edges comprising tuples of two end nodes of an edge. There are two common and significant features of citation graph: (a)

[1] www.facebook.com

[2] plus.google.com

[3] In the rest of the paper, the term 'venue' indicates either a journal or a conference.

it is directed and acyclic, and (b) it has a unidirectional growth, i.e. when it evolves over the time period, only new nodes and edges are added, and none of them are removed.

Co-authorship network, on the other hand, is another type of network that researchers have explored to study the amount of collaborations among the authors. It is also a simple graph $G^{'} = (X^{'}, Y^{'})$, where each node $x_i^{'} \in X^{'}$ represents an author and an undirected edge $y_{ij}^{'} \in Y^{'}$ between $x_i^{'}$ and $x_j^{'}$ indicates that the authors have published at least one paper together.

There are a few overlapping community detection techniques for unipartite networks such as Clique Percolation Method (CPM) [7] and iLCD [8], which can be applied on the citation or co-authorship networks individually. But, such approaches have the following disadvantages:

1) Any citation network is bounded by some fixed time interval, which indirectly discards several citation information of the older and newer papers.
2) The less cited papers of similar research areas are generally assigned into different communities by most of the community finding algorithms since they are sparsely connected and act as outliers.
3) In the case of co-authorship network, two authors working on the same research area should be assigned to the same community only if they have co-authored at least once, which may not always be possible.
4) Citation and co-authorship networks are usually sparse and disconnected which blur the utility of the general community detection algorithms.

In this paper, we consider all the problems mentioned above and propose an algorithm that can efficiently detect the overlapping communities in citation networks by utilizing both the citation and authorship information. To capture the multi-disciplinary research interests of the authors, journals/conferences, we create a tripartite “Publication Hypergraph” consisting of authors, papers and publication venues (journals/conferences) in three partitions with possibly unequal size.

Figure I shows the structure of publication hypergraph $H = (V, E)$ where the vertex set V constitutes three partitions V_A, V_P and V_J. Each hyperedge $e \in E$ connects a triple of nodes (a, b, c) where $a \in V_A$, $b \in V_P$ and $c \in V_J$. It indicates that the paper b written by the author a gets published in the venue c. It is important to note that if one paper is written by multiple authors then it is represented by multiple hyperedges having different nodes in author partition sharing common vertex in paper and venue partitions.

Some of the typical properties of the publication hypergraph are as follows:

- The size of each partition is possibly uneven with the relation $|V_P| \geq |V_A| \geq |V_J|$ follows in every publication hypergraph.
- There is an one-to-one relation between paper to journal, but the reverse may follow one-to-many relationship; and for author-to-paper and author-to-journal partitions, the relation is generally one-to-many.

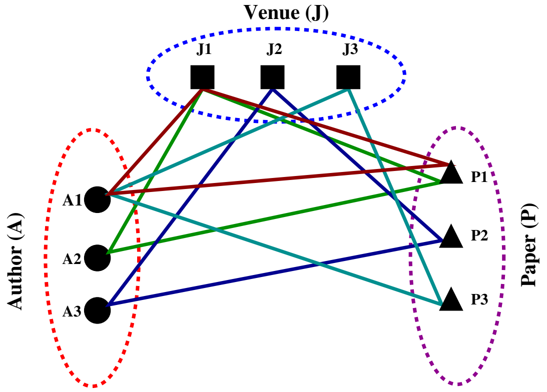

Fig. 1. Tripartite Publication Hypergraph

- Hyperedges that share a common vertex in paper partition, must share a common vertex in conference side and vice-a-versa.

Using this publication hypergraph representation and traditional citation network information, we develop an algorithm called *‘OverCite’* that clusters hyperedges of the publication hypergraph taking into account both hypergraph neighborhood and citation information which help detect the overlapping communities in the network. OverCite has the following advantages over other overlapping community detection algorithms applied only on the citation network.

1) OverCite can detect overlaps simultaneously from authors, papers and venues, which can not be possible by other unipartite overlapping detection algorithms.
2) It relies on both the co-authorship and citation information that can overcome the difficulty of assigning the older or newer papers into the appropriate communities.
3) It is independent of the size of the partitions; thus we can obtain different number of communities in different partitions.
4) The tripartite hypergraph representation of publication information can be easily extended to k-partite structure ($k > 3$) incorporating additional attributes in different partitions in order to get more reliable (hyper)links to detect communities.

We elaborately describe OverCite algorithm in Section II. We evaluate the performance of OverCite and compare it with other two state-of-the-art overlapping community detection algorithms (CPM and iLCD) in Section III. The experimental results show that OverCite outperforms other two algorithms. We develop a simple cluster based paper recommendation system, where the users can search papers using the paper title and with the returned search result, the system also recommends similar papers to users. The system is illustrated with the performance analysis in Section IV. In section V, we briefly narrate the related literature. Finally, we conclude the paper mentioning some important insights of this research with few future research directions in Section VI.

II. Our Proposed Algorithm: OverCite

This section details the proposed *OverCite* algorithm. OverCite detects overlapping communities of authors, papers and venues by using the publication hypergraph and the citation information simultaneously. OverCite follows a three-step procedure as described below.

First, OverCite converts the publication hypergraph H to its weighted line-graph $H^{'}$ where the hyperedges in H become nodes in $H^{'}$. Two nodes e_i and e_j in $H^{'}$ will be linked with a non-negative weight (representing their similarity) in terms of the following three factors: (a) Hypergraph Neighborhood Similarity (HNS), (b) Co-citation Strength (CCS) and (c) Bibliographic-Coupling Strength (BCS) which are described in this section later. The final similarity between e_i and e_j is measured by linearly combining HNS, CCS and BCS: $Similarity(e_i, e_i) = \alpha.HNS + \beta.CCS + \gamma.BCS$ (where, $0 \leq \alpha, \beta, \gamma \leq 1$).

Then, once the weighted line graph $H^{'}$ is constructed from the given tripartite hypergraph H, any community detection algorithm for weighted unipartite graph can be applied to cluster the nodes in $H^{'}$, that in turn produces communities of the hyperedges in H. We used Infomap algorithm [9][4], as it is proved to be very efficient algorithm to detect communities in large unipartite networks [10].

Finally, as the community structure is decided in $H^{'}$, each hyperedge in H (nodes in $H^{'}$) is assigned to a single community. This in turn assigns multiple overlapping communities to the nodes in H, since a node inherits membership of all those communities into which the hyperedges connected with this node are placed.

A. Similarity Metrics

We use the following three metrics for measuring similarity between the hyperedges that capture both the hypergraph neighborhood similarity and citation information based similarity to calculate the weight.

Hypergraph Neighborhood Similarity (HNS): Hypergraph Neighborhood Similarity computes the relative overlap between common neighbors of the end vertices of two hyperedges. Let $N^A(i)$, $N^P(i)$ and $N^J(i)$ denote the set of neighbors of node i of type V_A , V_P and V_J respectively (if $i \in V_A$, then $N^A(i) = \phi$, since nodes in the same partition are not linked). Similarity between two adjacent hyperedges $e_i = (a, b, c)$ and $e_j = (x, y, z)$ (where $a, x \in V_A$; $b, y \in V_P$; $c, z \in V_J$ and assumed $a = x$) is measured by the relative overlap among the neighbors of the non-common nodes of the same type:

$$HNS(e_i, e_j) = \frac{|S \cap S'| + |N^P(c) \cap N^P(z)| + |N^J(b) \cap N^J(y)|}{|S \cup S'| + |N^P(c) \cup N^P(z)| + |N^J(b) \cup N^J(y)|} \quad (1)$$

where $S = N^A(b) \cup N^A(c)$ and $S' = N^A(y) \cup N^A(z)$.

Co-citation Strength (CCS): Co-Citation Strength is measured by the number of times two papers are cited together in the subsequent literatures. It is an indication that both the papers treat related subject matter. The higher the co-citation is, the more citations the two papers have in common.

The relative measure of co-citation strength of two hyperedges e_i and e_j is defined by the ratio of the actual and maximum citations received by two end-points in paper partition. If $e_i = (a, b, c)$ and $e_j = (x, y, z)$ (where $b, y \in V_P$), $CCS(e_i, e_j)$ is defined as following

$$CCS(e_i, e_j) = \frac{|CITE(b) \cap CITE(y)|}{|CITE(b) \cup CITE(y)|} \quad (2)$$

where $CITE(b)$ is the set of papers which cite paper b. The range of CCS varies from 0 (when no one cites both the papers together) to 1 (when both papers are cited by the similar set of papers). Generally, the CCS value is maximum when both e_i and e_j have same end point in paper partition. As mentioned earlier, this is possible when multiple authors have written a paper and there are multiple entries of the author partition, sharing same vertex in the paper and venue partitions.

Bibliographic-coupling Strength (BCS): Bibliographic-Coupling Strength is defined by the number of common citations two papers mention in the reference sections. It is another way of determining the similarity between the related works of two papers. BCS between two hyperedges $e_i = (a, b, c)$ and $e_j = (x, y, z)$ will be computed using the end-points in paper partitions. If $b, y \in V_P$, BCS is defined as following

$$BCS(e_i, e_j) = \frac{|REF(b) \cap REF(y)|}{|REF(b) \cup REF(y)|} \quad (3)$$

where $REF(b)$ is the set of papers cited by paper b. The range of BCS also varies from 0 (when there is no citation common in the reference set of both papers) to 1 (when both papers cite same set of papers, or both are same paper).

The weighted line graph $H^{'}$ is generated by combining the three measures taking into account their relative importance in this context. The combined weight denoting the similarity of the nodes in the line graphs is: $(\alpha.HNS + \beta.CCS + \gamma.BCS)$, where α, β and γ are the relative weights (vary from 0 to 1) of the metrics. For large data, a subset of the data can be selected and optimal value for these parameters can be searched in the parameter space. Then the optimal values can be used for the larger dataset. The optimal value of the parameters may depend on the particular dataset the algorithm is applied to.

III. Evaluation

In this section, we evaluate the performance of OverCite and compare OverCite with the two popular overlapping community detection algorithms – Clique Percolation Method (CPM) and iLCD. The clique-percolation method [7] defines a community, or more precisely a k-clique community, as a union of all k-cliques (complete subgraphs of size k) that can be reached from each other through a series of adjacent k-cliques (where adjacency means sharing $k - 1$ nodes). This definition seeks to represent the fact that it is an essential feature of a community that its members can be reached through well-connected subsets of nodes. There are other parts of the whole network that are not reachable from a particular k-clique, but they potentially contain further k-clique communities. CPM searches for all k-cliques in the network, which in turn, assigns a single node to multiple communities. The particular implementation of CPM we used is 'CFinder' [11][5]. Note that, we use $k = 4$ since this is proved to be default parameter value in the literature [11].

[4] http://www.tp.umu.se/~rosvall/code.html

[5] www.cfinder.org

TABLE I. Description of Original and Filtered Dataset

	Original Dataset	Filtered Dataset
Number of valid indices	1,079,193	799,627
Number of entries with no venue	582	–
Number of entries with no author	5,773	–
Number of papers before 1960	886	–
Number of papers having no in-citation and out-citation	272,325	– –
Number of authors	662,324	495,311
Average number of papers by an author	3.82	3.52
Average number of authors per paper	2.615	2.609
Number of unique venue name	2,319	1,705

On the other hand, iLCD [8] is capable of detecting both static and temporal communities. Given a set of edges created at some time step, iLCD updates the existing communities by adding a new node if its number of *second neighbors* and number of *robust second neighbors* are greater than the expected values. New edges are also allowed to create a new community if the minimum pattern is detected. The similarity between two communities is defined as the ratio of nodes in common, and a merging procedure is performed to improve the detection quality if the similarity is high.

Now, we describe the real-world citation dataset and the metrics used to compare the algorithms. Later, the performance analysis on the dataset is presented with some interesting insights from the communities detected by OverCite.

A. Dataset Used

To create the publication hypergraph required for OverCite, only papers and their citation information are not adequate. We need several other related information about each paper, e.g. authors and publication venue. Additional information like year of publication, keywords and possibly a small abstract can help in evaluating the performance of the algorithms.

For our experiments, we used the dataset of computer science domain developed by Tang et al. [12][6]. It was constructed using the DBLP[7] web repository which contains information about various research papers from different fields of computer science domain published over the years. This information includes the title and authors of the research paper, index of the paper, year of publication, publication venue, list of research papers cited by the given paper and (in some cases) the abstract of the papers. The dataset is quite large with information of more than 1 million research papers.

We filtered the dataset to remove the papers with one or more missing entries (i.e. missing author names, missing publication venues etc). We also excluded reviews, surveys and text books from the dataset and only took contributory papers. Further, for fare comparison with community detection algorithms only using citation network, we considered only those papers that cite or are cited by at least one paper. This filtered dataset contains entries for around $700,000$ research papers. Table I presents the original and the filtered dataset in more detail.

TABLE II. Percentage of papers in various fields in computer science domain

No.	Subject	Abbreviation	% of papers
1.	Artificial Intelligence	AI	15.30
2.	Algorithms and Theory	ALGO	14.09
3.	Networking	NW	8.63
4.	Databases	DB	8.12
5.	Distributed and Parallel Computing	DIST	7.63
6.	Hardware & Architecture	ARC	7.29
7.	Software Engineering	SE	6.40
8.	Machine Learning and Pattern Recognition	ML	6.09
9.	Scientific Computing	SC	4.02
10.	Bioinformatics & Computational Biology	BIO	3.88
11.	Human-Computer Interaction	HCI	3.42
12.	Multimedia	MUL	3.34
13.	Graphics	GRP	3.32
14.	Computer Vision	CV	3.03
15.	Data Mining	DM	3.02
16.	Programming Languages	PL	3.00
17.	Security and Privacy	SEC	2.94
18.	Information Retrieval	IR	2.26
19.	Natural Language and Speech	NLP	2.11
20.	World Wide Web	WWW	1.76
21.	Computer Education	EDU	1.67
22.	Operating Systems	OS	1.07
23.	Real Time Embedded Systems	RT	0.90
24.	Simulation	SIM	0.14

B. Field Tagging

To develop a gold standard dataset for evaluating the performance of all three algorithms, we tag the field information of each paper using the Microsoft Academic Search Engine[8]. Microsoft Academic Search covers more than 48 million publications and over 20 million authors across a wide variety of domains. For papers in computer science domain, it categorizes them into 24 different fields with possibility of multiple fields assigned to a single paper. We crawled Microsoft Academic Search to find the field(s) of the papers present in the filtered dataset. Approximately, 88.12% of the papers could be tagged with their respective fields when searched with the paper title. Fields of rest 11.88% of the papers have been inserted using the conference/journal name where the paper has been published. Table II presents the percentage of papers in various fields in the tagged dataset.

Different fields can be thought of as different research areas. Papers having multiple fields are generally interdisciplinary in nature. Recently, Chakraborty et al. [13] showed that the intra-field citation density is generally much higher than the cross-field citation density. They proved using several community centric metrics [14] that the field information of the papers intrinsically indicates the natural communities of the citation network. Therefore, we have used the field information of the papers in the dataset as ground-truth communities and evaluated the performance of the algorithms against that. In the next subsection, we discuss the different metrics used for evaluating the algorithms.

C. Evaluation Metrics

To measure the accuracy of a clustering algorithm, the set of clusters found by an algorithm is compared against the set of *true* clusters. A normalized measure is desirable in many contexts, for example assigning a value of 0 where the two sets are totally dissimilar, and 1 where they are identical. We use the following three standard metrics to evaluate the algorithms.

[6] http://arnetminer.org/citation, named as *DBLP-Citation-network V4*

[7] www.informatik.uni-trier.de/~ley/db/

[8] http://academic.research.microsoft.com/

1) Rand Index: Rand Index [15] is a measure of the similarity between two data partitions. Given a set of n elements $S = \{O_1, O_2, ..., O_n\}$ and two partitions of S to compare ($X = \{X_1, X_2, ..., X_r\}$, a partition of S into r subsets; and $Y = \{Y_1, Y_2, ..., Y_s\}$, a partition of S into s subsets), Rand Index R can be defined as

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}} \quad (4)$$

where $a =$ the number of pairs of elements in S that are in the same set in X and in the same set in Y, $b =$ the number of pairs of elements in S that are in different sets in X and in different sets in Y, $c =$ the number of pairs of elements in S that are in the same set in X and in different sets in Y, and $d =$ the number of pairs of elements in S that are in different sets in X and in the same set in Y.

2) Omega Index: Omega Index [16] is the overlapping version of the Rand Index. It is based on pairs of nodes in agreement in two covers. Here, a pair of nodes is considered to be in agreement if they are clustered in exactly the same number of communities (possibly none). That is, the omega index considers how many pairs of nodes belong together in no clusters, how many are placed together in exactly one cluster, how many are placed in exactly two clusters, and so on. The detailed formulation of Omega Index can be found in [17].

3) ONMI: Aaron et al. [18] proposed a metric called 'Overlapping Normalized Mutual Information' (ONMI)[9] which takes into account the unintuitive behavior of Normalized Mutual Information (NMI) [19] and correct it by using a more conventional normalization. This metric has been reported outperforming the other metrics like Rand Index, Omega Index and the metric proposed by Lancichinetti et al. [20]. The detailed formulation of this metric can be found in [18].

D. Experimental Results

As mentioned earlier, we use the field information of the papers as ground-truth. Note that, around 30% of the papers in the dataset have multiple field tags, i.e. they actual belong to multiple research areas as identified by the Microsoft Academic Search engine. The field tag of a paper also indicates the related research area of the authors, and the related topics of the venue where it got published. In this way, we can identify the ground-truth community membership of each paper, author and venue present in our dataset.

Tuning the parameters for OverCite: We first create publication hypergraph and citation network from the dataset. We then use snowball sampling[10] on the large hypergraph to extract a smaller instance containing $10,000$ nodes. We randomly choose some seed nodes and the hypergraph grows by adding nodes which are connected to these nodes. The process is repeated until we get our required number of nodes.

Then OverCite is applied on the small hypergraph iteratively with different combinations of α, β and γ values and ONMI is computed for each iteration. The best parameter values in terms of the highest ONMI are reported as follows: $\alpha = 0.45$, $\beta = 0.32$ and $\gamma = 0.23$. We retain these values in all the experiments presented in the paper.

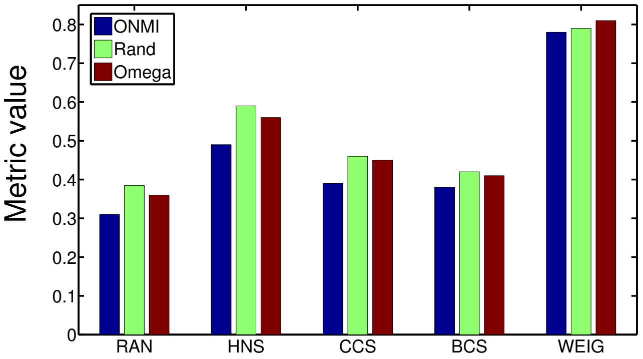

Fig. 2. Values of different evaluation metrics for the community structure obtained by assigning each of the hyperedges of H randomly into some community (RAN), the community structure obtained by using each of the measures separately for weight calculation and the actual weighted combination (WEIG) used in the entire analysis.

Comparing different similarity metrics used in OverCite: It would be interesting to analyse the significance of individual similarity metrics (described in section II) used to calculate the edge weights in the weighted line graph $H^{'}$. Furthermore, in order to show the community structure is not randomly found from the network, we assign each hyperedge of the publication hypergraph randomly into one of the 24 communities (since we know that there are 24 possible communities, i.e. 24 research fields present in the ground-truth community structure of the network). We plot the accuracy of the randomly assigned community structure and the communities obtained by using each of the three measures separately in Figure 2. We notice that the communities can not be detected randomly. Using only Hypergraph Neighborhood Similarity (HNS) performs better compared to using other metrics. However, the weighted combination of the three metrics overshadows all other metrics used individually.

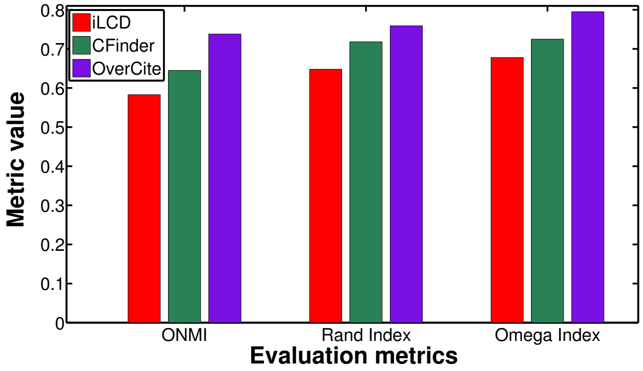

Fig. 3. Performance analysis of the three algorithms using three different evaluation metrics.

Comparing different community detection algorithms: We run all three algorithms (CFinder, iLCD and OverCite) on the dataset and compute the metrics ONMI, Rand Index and Omega Index using the ground-truth communities. Figure 3 plots the performances of the three different algorithms. As

[9] A C++ implementation is available at https://github.com/aaronmcdaid/Overlapping-NMI

[10] http://en.wikipedia.org/wiki/Snowball_sampling

shown in the figure, OverCite outperforms other two competing algorithms for all the metrics used in this experiment.

Amount of overlap detected by OverCite: We use OverCite to detect the overlapping communities from three partitions of the publication hypergraph. Figure 4 shows the distribution of overlapping vertices (the probability of a vertex belonging to multiple communities) in each partition. The propensity of the vertices belonging to multiple communities is higher in venue partition. This indicates that the journals/conferences now-a-days tend to become more interdisciplinary and the related topics of the venues spread into different areas of research. The probability of the paper belonging to different community is reasonably smaller than the author partition since generally a paper comprises not so diverse topic of interest, while an author has multiple research interests throughout her research career.

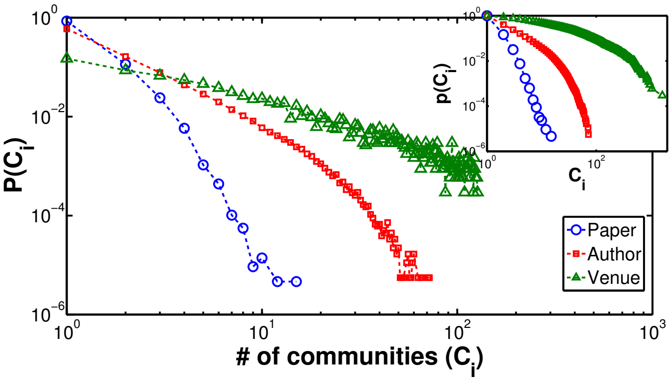

Fig. 4. Non-cumulative distribution of overlapping vertices in each partition (Inset: cumulative distribution of overlapping vertices).

Exploring the communities detected by OverCite: When analysing the results in more granular level, we observe some interesting insights. OverCite identifies some papers (examples shown in Table III) which are not assigned into single community by other algorithms. We notice that these papers constitute either at least one common author or they belong to same field or published in the same venue. Since they are not cited much, they were treated as outliers by other algorithms and assigned to some random communities. Contrary to these algorithms, OverCite takes into account the venue, author as well as the citation information when constructing the line graph. Therefore, in this case, such outliers are more likely to be assigned to the appropriate communities. Thus Overcite overcomes one of the shortcomings (addressed in section I) of the traditional overlapping community detection algorithms.

As mentioned earlier, one of the advantages of using OverCite is that it can detect overlapping communities simultaneously from papers, authors and venues. Therefore, it would be interesting to analyse the highly overlapped authors and venues obtained from OverCite. We tabulate top five highly overlapped authors and venues detected by OverCite in Table IV and search their corresponding fields of interest from Microsoft Academic Search. Interestingly, the research areas of each of the top five authors are largely diverse in nature. We notice that they are not the authors receiving highest citations in our dataset; still their works are multidisciplinary in nature. For the top four overlapped venues detected by OverCite, even Microsoft Academic Search could not assign distinct fields to them, i.e. they seem to be very general to the computer science domain.

TABLE IV. TOP OVERLAPPED AUTHORS AND VENUES IDENTIFIED BY OVERCITE. THE FIELD INFORMATION ARE EXTRACTED FROM MICROSOFT ACADEMIC SEARCH

		Fields
Authors	Mahmut Kandemir (Pennsylvania State University)	Hardware & Architecture, Dist. & Parallel Computing, Programming Language
	Gordon Blair (Lancaster University)	Networking, Dist. & Parallel Computing, Software Engineering
	Donald F. Towsley (University of Massachusetts)	Networking, Operating System, Multimedia
	Ricardo A. Baeza-yates (Yahoo Research Labs)	Information Retrieval, Algorithms and Theory, World Wide Web
	Mary Lou Soffa (University of Virginia)	Software Engineering, Programming Languages, Dist. & Parallel Computing
Venues	Communications of the ACM	Computer Science
	IEEE Computer	Computer Science
	Journal of the ACM	Computer Science
	PIEEE (Proceedings of The IEEE)	Computer Science
	DAC (Design Automation Conference)	Hardware & Architecture

IV. Recommendation System

To test the effectiveness of different community detection algorithms in a real-world application setting, we designed a simple search and paper recommendation system. Here, users can search papers using the paper title and with the returned search result, the system also recommends similar papers to users. The recommendation is purely cluster-based, i.e. depending on the paper searched, other papers of its containing cluster form the recommendation set. We ran three different community detection algorithms (OverCite, iLCD and CFinder) to find the paper clusters and then those detected clusters were used for recommendation purposes.

Once the recommendation system was ready, we asked volunteers to evaluate the recommendations provided by different community detection algorithms. 38 students of computer science department of a particular institute participated. Total 207 unique papers were searched. For each paper, the recommendation system showed recommendations by all three algorithms and volunteers were asked to tag each recommended paper as Relevant or Non-relevant. Total 3612 relevance judgments were received. Then we used standard Information Retrieval metric 'Precision' [19] to compare the performance of the three different algorithms.

Precision (P): Precision is the fraction of retrieved documents that are relevant.

$$Precision = \frac{Number\ of\ relevant\ items\ retrieved}{Number\ of\ retrieved\ items} \quad (5)$$

Figure 5 shows the average precision value for all the algorithms. It is evident from the figure that OverCite outperforms all other algorithms in recommending useful papers to the users.

TABLE III. Selected pairs of papers that are assigned into the same community by OverCite, not by other two comparative algorithms.

• S. Ferilli, F. Esposito, T.M.A. Basile and N.D. Mauro. Automatic Induction of Domain-Related Information: Learning Descriptors Type Domains, *ECAI*, 2004. • N. D. Mauro, F. Esposito, S. Ferilli and T.M.A. Basile. A Backtracking Strategy for Order-Independent Incremental Learning, *ECAI*, 2004.
• B.J. Thibodeau, S.W. Hart, D.R. Karuppiah, J. Sweeney and O. Brock. Cascaded Filter Approach to Multi-objective Control, *ICRA*, 2004. • Y. Yang and O. Brock. Adapting the Sampling Distribution in PRM Planners based on an Approximated Medial Axis, *ICRA*, 2004.
• Maurizio Montagnuolo and Alberto Messina. Multimodal Genre Analysis Applied to Digital Television Archives, *DEXA Workshops*, 2008. • Pierre Allard and Sébastien Ferré. Dynamic Taxonomies for the Semantic Web, *DEXA Workshops*, 2008.
• Hung-Lung Wang, Bang Ye Wu and Kun-Mao Chao. The backup 2-center and backup 2-median problems on trees, *Networks*, 2009. • Mindaugas Bloznelis, Jerzy Jaworski and Katarzyna Rybarczyk. Component evolution in a secure wireless sensor network, *Networks*, 2009.
• Shripad Kondra and Vincent Torre. Texture Classification Using Three Circular Filters, *ICVGIP*, 2008. • Jean-Michel Morel,Philippe Salembier. Monocular Depth by Nonlinear Diffusion, *ICVGIP*, 2008.

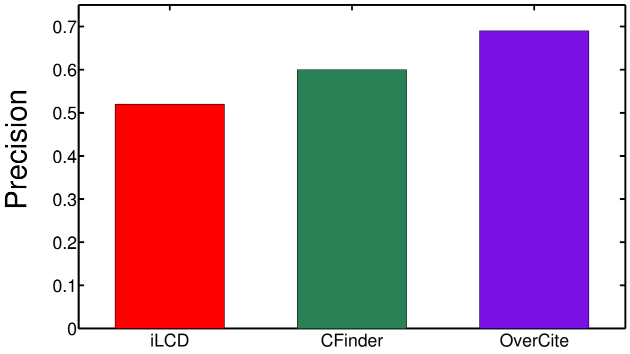

Fig. 5. Precision values for paper recommendation using different algorithms.

V. Related Work

Citation analysis research dates back to 1961 when the Science Citation Index began publication. Automated algorithms as used by CiteSeer[11], Google Scholar[12] etc. made it much more versatile and widespread. There are different applications of citation analysis. Systems like ArnetMiner [12] provides ranking of authors and publication venues. Yang et al. [14] considers citation network as multi-type network with integrated citations among papers, authors, affiliations and publishing venues in a single model. Zhao et al. [21] studies the relationship between authors using community mining techniques.

Community detection in its actual definition really began with the experiment of Girvan and Newman algorithm [22]. Since then, a large amount of algorithms have been proposed, sometimes with great improvements in time and efficiency. The experiment of Fortunato [10] confronted the best-known algorithms, proposing a benchmark (the LFR benchmark) that generates graphs with well-defined communities. According to his results, two algorithms, Infomap [23] and the fast modularity optimization by Blondel et al. [24], are the best algorithms available until now for unipartite networks. But they fail to detect overlaps across communities.

Several algorithms have been proposed for detecting communities in hypergraphs. Vazquez [25] proposed a Bayesian formulation of the problem of finding hypergraph communities. They start from a variational function which resolves the population structure by determining the hypergraph communities and model parameters from the data. The final Variational Bayes (VB) algorithm is a self-consistent set of equations for determining the group assignments and the model parameters. The VB algorithm is based on recursive equations similar to those for the *Expectation Maximization* (EM) algorithm. Lin et al. [26] proposed a multi-tensor factorization method for detecting hypergraph communities. All the hypergraph community detection algorithms stated above assign a single community to each node.

Recently, k-clique percolation method (CPM) [27] attempted to address the overlapping community detection issue. Though it was reported as inefficient algorithm in terms of resource and time utilization [8], it provides the first and powerful step towards overlapping community detection. More efficient study on disjoint and overlapping community detection in social networks (undirected, directed and weighted), named as ‘SLPA’, has been proposed by Xie et al. [28], which is a general speaker-listener based information propagation process. Cazabet et al. [8] introduced iLCD (intrinsic Longitudinal Community Detection) that can detect dynamic communities of the networks in a longitudinal framework.

Some recent studies have proposed a few overlapping community detection algorithm for hypergraphs. Ghosh et al. [29] extended link clustering algorithm for graphs to hypergraphs. Chakraborty et al. [30] [31] proposed a method to detect overlapping communities in folksonomies, involving user, resource and tag nodes in a hypergraph. Recently, Du et al. [32] proposed a novel approach to identify overlapping for bipartite network. A few other techniques also have been developed for multipartite networks. However, to the best of our knowledge, Overcite is the first algorithm to simultaneously utilize tripartite publication hypergraph and the unipartite citation network for detecting overlapping communities of authors, papers and publication venues.

[11] citeseerx.ist.psu.edu

[12] scholar.google.com

VI. Conclusion

In this paper, we propose an efficient algorithm 'OverCite' to detect overlapping communities in citation network. For that, we construct a publication hypergraph constituting authors, papers and publication venues in three partitions, and convert it into weighted line graph after including the citation information. Then communities are detected in the weighted line-graph which in turn produces overlapping communities of authors, papers and publication venues simultaneously.

OverCite not only outperforms two state-of-the-art overlapping community detection methods, but also explores several significant insights that can be indicative to judge the efficiency of the algorithm. We also develop a simple recommendation system, which systematically utilizes the overlapping community structure of the citation network, and shows recommendation to users. However, this recommendation system needs to be improved further. In future, we plan to develop a more efficient recommendation system that can simultaneously consider collaborative filtering of users and community structure of the network. We are also interested to see the performance of OverCite after including more partitions (publication year, keywords of the paper etc.) in the hypergraph.

VII. Acknowledgements

The authors would like to thank Ayushi Dalmia, Heritage Institute of Technology, India for assisting the experiments as well as for interesting discussions on the subject. The first author of the paper is financially supported by the Google India PhD fellowship Grant in Social Computing.

References

[1] S. Fortunato, "Community detection in graphs," *CoRR*, 2009.

[2] C. L. Giles, K. D. Bollacker, and S. Lawrence, "Citeseer: an automatic citation indexing system," in *Proceedings of the third ACM conference on Digital libraries*, ser. DL '98. ACM, 1998, pp. 89–98.

[3] J. E. Hirsch, "An index to quantify an individual's scientific research output that takes into account the effect of multiple coauthorship," *Scientometrics*, vol. 85, no. 3, pp. 741–754, Dec. 2010.

[4] E. Garfield, I. Sher, and R. Torpie, *The Use of Citation Data in Writing the History of Science*. Inst. for Scientific Information Incorporated, 1964.

[5] R. Guimera, B. Uzzi, J. Spiro, and L. Amaral, "Team Assembly Mechanisms Determine Collaboration Network Structure and Team Performance," *Science*, vol. 308, no. 5722, pp. 697–702, 2005.

[6] A. M. Odlyzko, "Tragic Loss or Good Riddance? The Impending Demise of Traditional Scholarly Journals," *Journal of Universal Computer Science*, vol. 0, no. 0, pp. 3–52, 1994.

[7] G. Palla, I. Dernyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, June 2005.

[8] R. Cazabet, F. Amblard, and C. Hanachi, "Detection of Overlapping Communities in Dynamical Social Networks," Aug. 2010, pp. 309–314.

[9] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 4, pp. 1118–1123, 2008.

[10] S. Fortunato and A. Lancichinetti, "Community detection algorithms: a comparative analysis: invited presentation, extended abstract," in *Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools*. ICST, 2009.

[11] B. Adamcsek, G. Palla, I. J. Farkas, I. Derényi, and T. Vicsek, "Cfinder: locating cliques and overlapping modules in biological networks," *Bioinformatics*, vol. 22, no. 8, pp. 1021–1023, 2006.

[12] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: extraction and mining of academic social networks," in *ACM SIGKDD*, 2008, pp. 990–998.

[13] T. Chakraborty, S. Sikdar, V. Tammana, N. Ganguly, and A. Mukherjee, "Computer science fields as ground-truth communities: Their impact, rise and fall," in *the proceedings of ASONAM*, 2013 (accepted).

[14] Z. Yang, L. Hong, and B. D. Davison, "Topic-driven multi-type citation network analysis," in *Adaptivity, Personalization and Fusion of Heterogeneous Information*, ser. RIAO '10, 2010, pp. 24–31.

[15] W. M. Rand, "Objective Criteria for the Evaluation of Clustering Methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.

[16] L. M. Collins and C. W. Dent, "Omega: A general formulation of the rand index of cluster recovery suitable for non-disjoint solutions," *Multivariate Behavioral Research*, vol. 23, no. 2, pp. 231–242, 1988.

[17] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: the state of the art and comparative study," *CoRR*, 2011.

[18] A. F. McDaid, D. Greene, and N. J. Hurley, "Normalized mutual information to evaluate overlapping community finding algorithms," *CoRR*, 2011.

[19] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press, 2008.

[20] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, 2009.

[21] Q. Zhao, S. S. Bhowmick, X. Zheng, and K. Yi, "Characterizing and predicting community members from evolutionary and heterogeneous networks," in *Proceedings of the 17th ACM conference on Information and knowledge management*, ser. CIKM '08. New York, NY, USA: ACM, 2008, pp. 309–318.

[22] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[23] M. Rosvall and C. Bergstrom, "An information-theoretic framework for resolving community structure in complex networks," *Proceedings of the National Academy of Sciences*, vol. 104, no. 18, 2007.

[24] V. D. Blondel, J. loup Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, no. 18, 2008.

[25] A. Vazquez, "Finding hypergraph communities: a Bayesian approach and variational solution," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, Jul 2009.

[26] Y.-R. Lin, J. Sun, P. Castro, R. Konuru, H. Sundaram, and A. Kelliher, "Metafac: community discovery via relational hypergraph factorization," in *Proc. ACM SIGKDD Conference*, 2009, pp. 527–536.

[27] F. Breve, L. Zhao, and M. Quiles, "Uncovering overlap community structure in complex networks using particle competition," in *Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence*, ser. AICI '09. Springer-Verlag, 2009, pp. 619–628.

[28] J. Xie, B. K. Szymanski, and X. Liu, "Slpa: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process," in *ICDM 2011 Workshop on DMCCI*, 2011.

[29] S. Ghosh, P. Kane, and N. Ganguly, "Identifying overlapping communities in folksonomies or tripartite hypergraphs," in *Proc. ACM Conference on World Wide Web (WWW) companion volume*, Mar 2011, pp. 39–40.

[30] A. Chakraborty, S. Ghosh, and N. Ganguly, "Detecting overlapping communities in folksonomies," in *ACM HyperText*, 2012, pp. 213–218.

[31] A. Chakraborty and S. Ghosh, "Clustering Hypergraphs for Discovery of Overlapping Communities in Folksonomies," *Dynamics On and Of Complex Networks Applications to Biology, Computer Science, and the Social Sciences*, vol. 2, 2013.

[32] N. Du, B. Wu, B. Wang, and Y. Wang, "Overlapping Community Detection in Bipartite Networks," 2008.