Predicting New Collaborations in Academic Citation Networks of IEEE and ACM Conferences

Irfan A. Shah, Muhammad U. Ilyas, Mamoon Raja, Saad Saleh,
Muhammad Murtaza Khan, Ali Mustafa Qamar
School of Electrical Engineering and Computer Science
National University of Sciences and Technology, H-12, Islamabad - 44000, Pakistan
{09mscsemshah, usman.ilyas, mamoon.raja, saad.saleh,
muhammad.murtaza, mustafa.qamar}@seecs.edu.pk

M. Zubair Shafiq, Alex X. Liu, Hayder Radha
College of Engineering, Michigan State University, East Lansing, MI - 48823, USA
{shafiqmu, alexliu}@cse.msu.edu, radha@egr.msu.edu

Abstract

In this paper we study the time evolution of academic collaboration networks by predicting the appearance of new links between authors. The accurate prediction of new collaborations between members of a collaboration network can help accelerate the realization of new synergies, foster innovation, and raise productivity. For this study, the authors collected a large data set of publications from 630 conferences of the IEEE and ACM of more than $257,000$ authors, $61,000$ papers, capturing more than $818,000$ collaborations spanning a period of 10 years. The data set is rich in semantic data that allows exploration of many features that were not considered in previous approaches. We considered a comprehensive set of 98 features, and after processing identified eight features as significant. Most significantly, we identified two new features as most significant predictors of future collaborations; 1) the number of common title words, and 2) number of common references in two authors' papers. The link prediction problem is formulated as a binary classification problem, and three different supervised learning algorithms are evaluated, *i.e.* Naïve Bayes, C4.5 decision tree and Support Vector Machines. Extensive efforts are made to ensure complete spatial isolation of information used in training and test instances, which to the authors' best knowledge is unprecedented. Results were validated using a modified form of the classic 10-fold cross validation (the change was necessitated by the way training, and test instances were separated). The Support Vector Machine classifier performed the best among tested approaches, and correctly classified on average more than 80% of test instances and had a receiver operating curve (ROC) area of greater than 0.80.

1 Introduction

1.1 Background and Motivation

This paper presents a Machine Learning approach to predict new, future collaborations in an academic collaboration network between researchers that have not collaborated previously. A collaboration between two authors is classified as new at time "t" if the authors have not collaborated to co-author a paper before time "t" but will do so after time "t". The successful prediction of future collaborators can help raise productivity and potentially lead to more fruitful, mutually beneficial professional relationships between researchers. While we review several prior works that have attempted to understand the process by which new links are created in social networks, the process is not very well understood and may differ in different contexts.

1.2 Limitations of Prior Art

A number of previous studies have focused on the link prediction of nodes based upon the behaviour in the past. However, performance of previous schemes was limited because of various reasons. Firstly, several studies have focused over the set of features for authors that were limited to topological properties of nodes, that were representing authors in a graph [18]. On the contrary, our approach is a generic approach such that it can be applied to any social network graph, *i.e.* the method itself does not make any prior assumptions that require the social network to necessarily be an academic collaboration network. Secondly, several studies have studied over the semantic features along with topological properties to predict links in networks [24] [2]. These studies have derived semantic features from texts, and documents associated with authors. They explored features for the prediction of link appearance, explicitly for co-authorship networks, and implicitly for general social networks. Moreover, majority of these studies are not clear about whether they ensured complete disjointness between data points in training and test data [24].

1.3 Proposed Solution

We cast the link prediction problem as a supervised, binary classification problem. We explored topological as well as semantic features for prediction. However, the richness of the data set that was collected allowed the exploration of many more features of both types. Moreover, to the authors' best knowledge, we took unprecedented steps to ensure mutual exclusivity between data points used in training and test data sets, as well as between folds in the evaluation phase.

ISBN: 978-1-62561-000-3

1.4 Results and Findings

Out of the 98 features considered, the eight most significant were identified and used for prediction. Link prediction is formulated as a binary classification problem and three different supervised learning algorithms were evaluated, *i.e.* Naïve Bayes, C4.5 decision tree and support vector machines (SVM). Extensive efforts were made to ensure complete spatial isolation of information used in training and test instances, which to the authors' best knowledge is unprecedented. Results were validated using a modified form of the classic 10-fold cross validation (the change was necessitated by the way training and test instances were separated). The SVM classifier was determined to provide best performance with a receiver operating curve (ROC) area greater than 0.80. It correctly classified on average more than 80.9% of test instances it was provided. Our findings show that high degree of overlap in previous publications is an even more significant predictor of new collaborations than the proximity of two researchers in collaboration network. The baseline approach by Al Hassan *et al.* has an accuracy of 66.5% (assuming equal prior), compared to an accuracy of 81%.

1.5 Key Contributions

The contributions of this research study are three-fold.

1. Our principal contribution is the formulation of the link prediction problem as a binary classification problem on a dynamic graph, and the subsequent development and evaluation of multiple machine learning classifiers that predict future collaborations with high accuracy.

2. Our second contribution is the identification of eight most significant joint features of pairs of authors on the basis of which link prediction can be performed. Most significantly, we observed that the number of words common between the set of words used by authors in their papers title, and the number of cited references common between their papers are the two most significant features among all the features we considered.

3. Our third contribution is the collection of an original data set consisting of information about publications that appeared in flagship conferences of all IEEE Societies and ACM special interest groups (SIG) in the $10+$ year period from 2001 to 2011.[1] This information includes author names, paper title, keywords, abstract (when available), references / citations, conference name, and publication date.

Paper Organization: The rest of this paper is organized as follows: The *Related Work* section summarizes the previous approaches to link prediction in academic citation networks. The *Data Set* section is a description of the original data set collected for this work, and provides some basic descriptive statistics. The *Problem Formulation* section describes the problem formulation. The *Methodology* section describes the method development of various classifiers. The *Results* section shows the results of link prediction using various features and classifiers and discusses them. The *Conclusions* section summarizes and concludes the paper.

2 Related Work

The evolution of the social networks can be divided into two broad categories: Microscopic evolution and analysis and Time evolving structural analysis. The microscopic evolution and analysis covers the aspects which follow the Power law distribution encompassing the addition (or removal) of nodes in a graph, or the addition (or removal) of edges between nodes. Leskovec *et al.* [16] proposed such a model with known node and edge arrival rates and lifetimes.

Early work by Krebs [13] used topological properties such as clustering coefficient, mean path lengths, degree, betweenness centrality and closeness centrality to map the collaboration network of the $9/11$ terrorists. Krebs concluded that the tie strength of the edges should be measured to distinguish between criminal and non-criminal elements,

Al Hasan *et al.* [2] formulated a generic framework for link prediction. They evaluated a list of features as predictors that included both topological as well as semantic features on the BIOBASE [5] and DBLP [17] datasets using nine and four features, respectively. Performance was reported for a list of eight different classifiers, and testing was done using 5-fold cross validation.

Liben-Nowell and Kleinberg [18] explored graph distances between pairs of nodes. The authors developed methods for link prediction based on the proximity of nodes in networks, and deduced that predictions can be made based on topological features. They provided a detailed analysis of the link prediction problem in social networks using the collaborative networks covering the methods formulated for link prediction including measures such as Adamic / Adar [1], Katz coefficient [12], Jaccard's coefficient [21] etc. While these predictors produced up to approximately 55 fold improvement in prediction accuracy relative to a random predictor, that translates to a best absolute accuracy of only 16% (using Katz clustering).

Tylenda *et al.* [23] emphasized the use of temporal information in favor of a static snapshot of graph. They extended the work by Wang *et al.* [24] that incorporated the temporal information in a probabilistic model. It included the use of weighted edges (with weights derived from temporal information) in proximity measure based prediction methods such as rooted PageRank and Adamic / Adar. They formulated the link prediction problem as a node ranking problem, and reported results in terms of Discounted Cumulative Gain and Average Normalized Rank.

Clauset *et al.* [7] inferred the hierarchical structure from networks to predict links. The authors show that the hierarchical properties of the networks can explain, and reproduce many commonly observed topological properties of the network thereby aiding the prediction of links in the future. The authors show this by modeling the network data, and using the statistical inference along with maximum likelihood approach, and using the Monte Carlo sampling algorithm on all probable dendograms.

Lee *et al.* [14] proposed a graph modification process for link prediction in heterogeneous bibliographic network. Based upon the new graph, authors implemented random walk-based algorithm to compute critical links. The data set used contains only 2505 authors, limited over DBLP. Moreover, developed approach follows an iterative technique which takes significant computation time, and cannot be parallelized.

Sun *et al.* [22] developed a novel method called $PathPredict$

[1] The data set will be made public after publication.

for co-authorship prediction in heterogeneous bibliographic networks. At first stage, meta-path based features are separated from the bibliographic network. At second stage, supervised learning technique is employed to weigh the various links between authors. A real data set rich in semantics is collected from DBLP. Authors have concluded that meta-path based heterogeneous features can provide more accuracy as compared to homogeneous features.

More recently, Benchettara *et al.* [3] studied the link prediction problem in bipartite graphs, and used a dyadic topological approach. The data-set used was co-authorship data from DBLP bibliographical files, and an e-commerce website that makes product recommendations. The data was modeled as a bipartite graph, and projected into a uni-modal (uni-nodal) graph on which predictions were performed using topological, neighborhood, and distance based attributes. The authors extracted direct and indirect attributes using the bipartite graph and the projected (uni-nodal) graphs that provided a likelihood of a link between two nodes. Although it was concluded that using both direct and indirect attributes increased precision, the study lacked a comparative analysis with pre-existing techniques. Benchettara *et. al* used semantic features along with topological features but did not consider temporal features.

Radev *et al.* [20] presented a manual method to curate the network database of citations and collaborations for the association for computation linguistics (ACL) anthology. ACL anthology previously lacked the tendency to provide citation information. They claimed that citing sentences can be used to analyze the dynamics and trends of research.

Lichtenwalter *et al.* [19] employed both supervised and unsupervised machine learning techniques to predict links in sparse networks. They also proposed a new unsupervised, flow based prediction algorithm called *PropFlow*. They provided valuable insight as to how to frame a particular prediction problem, and why a supervised or unsupervised framework would be apt for the task. The authors' stated that as supervised learning schemes are capable of dealing with data sets with great class imbalances, and focusing on the class boundaries whereas unsupervised learning techniques cannot combat imbalance. The authors suggested that for networks with convergence and saturation issues, the training period should be as long as possible, whereas for networks like the Internet where the snapshots contain the entire network, such requirements are not necessary. The provided feature list included detailed properties extracted from multi-graphs, but did not mention any semantic information. It can be concluded that classification was carried out on topological features only.

3 Data Set

Pre-existing data sets in the public domain contain very limited information. The Stanford Network Analysis Project (SNAP) [15] hosts several such data sets. However, most citation network data sets often do not contain timestamp information, and/or contain very limited information about the publications produced by researchers. This drastically reduces the search space for useful features used for prediction, and often times limits them to graph theoretic measures of node distances. As we will also show, these data sets are one to two orders of magnitude smaller relative to the dataset collected as part of this study.

Qty.	**IEEE + ACM [This paper]**	**BIO-BASE** Al Hasan *et al.*	**DBLP** Al Hasan *et al.*	**DBLP** Tylenda *et al.*	**Astro-ph** Tylenda *et al.*	**Astro-ph** Liben-Nowell *et al.*	**Cond-mat** Liben-Nowell *et al.*	**ACL** Radev *et al.*
Nodes	325,268							
Authors	257,565	156,561	1,564,617	437,515	55,233	5,343	5,469	14,799
Confs.	630							
Papers	61,393	831,478	540,459	522,932	60,996	5,816	6,700	18,290
Timespan (in yrs)	10			10	10	3	3	
Edges	3,656,375							
Collabs.	818,596			1,359,471	644,496	41,852	19,881	
Citations	2,714,993							84,237

Table 1: Data set sizes.

To evaluate new features for predicting the emergence of new collaborations between researchers, the need was felt for collecting a large data set that contains the data necessary to compute these new features. We targeted the electrical engineering and computer science community of researchers. We collected conference paper information of all flagship conferences of 38 IEEE Societies and 32 ACM SIGs for the 10 year period from 2001 to 2011. The data set contains $61,393$ research articles, and $257,565$ authors in approximately 630 conferences. The data set further contains various interactions between authors, research articles, and other authors. Table 1 provides basic numbers describing the raw data set as well as the graph built using it.

Data was collected by means of a web crawler. Data collected for each paper included paper title, author name(s), list of keywords, abstract, year of publication, and cited papers. Data collected by the crawler was stored in flat plain text files (one for each conference) in Python dictionary format for further use.

Our goal is to predict collaborations between authors. One of the key challenges of using publication information from different sources was the numerous format variations of names and references. For example, among the different formats of names we came across were; '*<first name> <last name>*', '*<last name>, <first name>*', '*<first initial>. <last name>*', '*<first name> <middle initial>. <last name>*' etc. We standardized author names to the '*<first initial>. <middle initial>. <last name>*' format (used by IEEE in its IEEE Xplore database). This approach is identical to the approach used to standardize names by Liben-Nowell and Kleinberg [18].

Each reference cited in a paper is parsed for author name(s) (and standardized according to the preceding name format), title, conference name, publication year. Many cited papers are not listed in the period or the conferences we targeted for collection. This suggests that they are not accompanied by keywords and abstracts. For these papers we created our own set of keywords by removing stopwords from their title string using the Python NLTK (Natural Language Toolkit) [4] module.

3.1 Graphical Modeling

We model the collected data as a directed multigraph $G(V, E)$ consisting of the set of vertices V and set of edges E. Vertices or nodes in the set V are one of three types depending on what they represent; paper nodes, author nodes, and conference nodes. Similarly, the types of edges in the set E vary according to the types of nodes at their end points. We make a directed graph, called the "Parent graph" that incorporates all data from the data set. Other, more simplified graphs representing a subset of the available data are used for the computation of some features. The nodes in the *Parent graph* may be one of three basic types; paper nodes, author nodes, and conference nodes. We will now list the types of edges that are found in the Parent graph.

Paper-Author Edge: In the Parent graph, each paper node is connected by a directed edge from author to paper node representing an author of the paper. We will refer to such edges as *paper-author* edges. Every paper is linked via an edge to each of its author nodes.

Paper-Conference Edge: Each paper is also connected by a directed edge to the conference node representing the conference, and year it was published in. We will refer to such edges as *paper-conference* edges. The resolution of the temporal information is 1 year, and is stored in the form of the year the conference was held. It is frequently the case that a paper cites another paper that was published in a conference / journal that was not covered by the crawler. In such cases, we parse the name of the conference from the available bibliography as best as we can, and create a node for that conference as well. When the name of a conference is unrecognizable we assign such cited papers to an *Other* conference node.

Paper-Paper Edge: Finally, citations are represented by a directed edge from the citing paper to the cited paper. We will refer to such edges as *paper-paper* or *citation* edges.

Author-Author Edge: As mentioned earlier, the parent graph incorporates all data in the data set either as structural information or as node/edge attributes. However, for the computation of certain features, it is easier to work with alternative graph representations based on a subset of the data. One such frequently used graph (that will be described shortly) is what we call the *Author-author* graph. The author-author directed multigraph contains a fourth kind of edge that we call the author-author edge. An author-author edge can represent one of two different possible relationships between authors; collaboration/co-authorship, and citation. To differentiate between these two different kinds of relationships, author-author edges in the author-author graph are colored, red for collaborations (two directed edges from one author to the other and vice versa) and blue for citations (from citing to cited author). The parent graph can be projected from its Author-Paper-Conference-Time space to obtain a flattened representation in a lower dimensional subspace. This flattening is performed to simplify the computation of certain features. The various flattened graphs that we employed to facilitate computation of features are listed here. Simple illustrative examples are shown in Fig. 1. All the following graphs were modeled and node features were computed in Python using the NetworkX module [10].

Author-Author Graph: A graph containing only author nodes, and (directed) author-author edges colored either red or blue. Red edges represent collaborations/co-authorship of a paper, while blue edges represent citations. Most public citation network data sets today are author-author graphs.

Author-Paper Graph: A graph containing only author, and paper nodes. This graph is obtained by simply removing all conference nodes from the parent graph.

Author-Conference Graph: A graph containing only author, and conference nodes. This graph is used to simplify determination of overlap in publication venues between two authors.

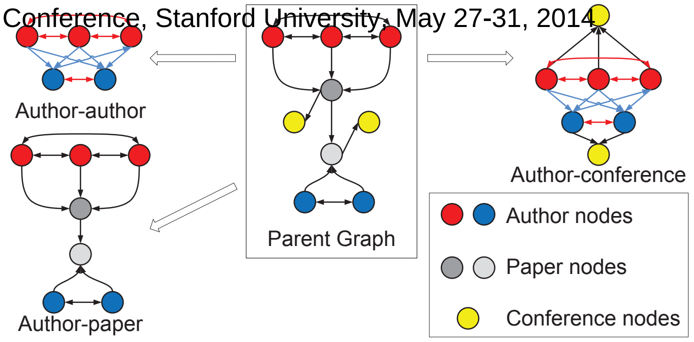

Figure 1: Projections of the parent graph from the Author-Paper-Conference-Time space to lower dimensional subspaces.

4 Problem Formulation

The link prediction problem can be formally described as follows; Given a dynamic graph $G(V, E, t_1, t_2)$ consisting of vertices or nodes V (representing authors, papers, and publication venues), and edges E (links between the nodes) representing timestamped authorship, citation, and publication relationships between authors, papers, and conferences during time t_1, and t_2, we aim to predict the appearance of edges representing new collaborations between authors during time t_2, and t_3 (where $t_1 < t_2 < t_3$). The predictor uses links recorded between times t_1, and t_2, the training interval, and the formulated algorithm to give accurate prediction of links that are likely to appear in the future time interval from t_2 to t_3, the test interval. We propose to design a binary classifier. For each inspected pair of previously non-collaborating authors, and based on their history over the period of time $[t_1, t_2]$, the classifier will determine whether or not they will collaborate in the period $[t_2, t_3]$.

5 Methodology

5.1 Data Partitioning

The link prediction problem is formulated as a supervised binary classification problem. For this reason the data set was divided into training and test data in the manner described in this section.

Wang *et al.*'s [24] clearly shows that their selection of training and test data is neither disjoint in time, nor in space. Liben-Nowell and Kleinberg [18] and Tylenda *et al.* [23] both ensured that training and test data sets are disjoint in time. However, the descriptions of neither works assure disjointness in space.

Unlike these previous approaches, we ensured that training and test data sets are disjoint in space, *i.e.* authors that are part of the training set may not be included in the test set (and vice versa). Since many of the features of pairs of authors are mathematical functions of node properties, the node disjoint data sets ensure that instances in the training set do not bias the design of the classifier (and detection performance) for the test set. Fig. 2 depicts a small illustrative example that shows how this is achieved. The complete data set spans the period 2002 to 2011, both years included. Data from the 2002 to 2006 time period,

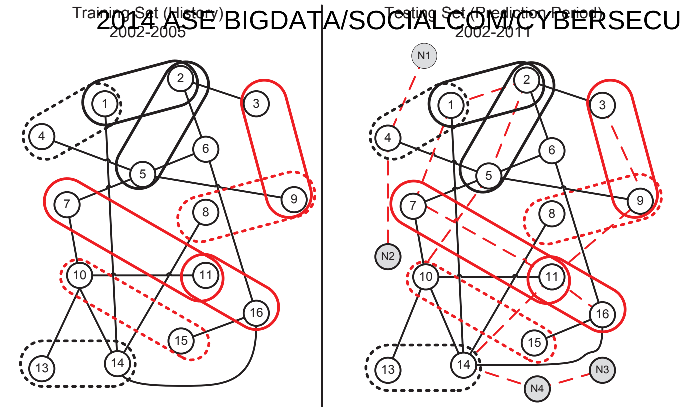

Figure 2: Temporally and spatially disjoint partitioning of data into training and test sets. Depending on the kind of graph, nodes can represent authors, papers or conferences, and the links between them represent various associations described in the subsection *Graphical Modeling*. N1, N2, N3 and N4 are nodes that were previously not present in the training interval and only appear in the test interval.

Author property		**F1**	**F2**	**F3**	**F4**	**F5**	**F6**	**F7**	**F8**	**F9**
Topological Features										
Degree	**P1**	✔	✔	✔	✔		✔		✔	✔
Squ. Clustering	**P2**	✔	✔	✔	✔		✔		✔	✔
Closeness	**P3**	✔	✔	✔	✔		✔		✔	✔
Avg. Neighbor Degree	**P4**	✔	✔	✔	✔		✔		✔	✔
Triangles	**P5**	✔	✔	✔	✔		✔		✔	✔
Clustering Coefficient	**P6**	✔	✔	✔	✔		✔		✔	✔
Semantic Features										
No. of Keywords	**P7**	✔	✔	✔	✔		✔		✔	✔
KLD of Keywords	**P8**	✔	✔	✔	✔		✔	✔	✔	✔
Entropy of Keywords	**P9**	✔	✔	✔	✔		✔	✔	✔	✔
No. of Words in Titles	**P10**	✔	✔	✔	✔	✔	✔		✔	✔
No. of Words in Conf.	**P11**	✔	✔	✔	✔	✔	✔		✔	✔
No. of Collaborators	**P12**	✔	✔	✔	✔	✔	✔		✔	✔
No. of Cited Authors	**P13**	✔	✔	✔	✔	✔	✔		✔	✔

Table 2: Author pair features.

also referred to as *history*, is flattened to produce a graph used for the computation of features. Based on these features, it is predicted whether or not a given pair of authors that were both present in the 2002 to 2006 period, but did not collaborate then, will collaborate in the observable time period following 2006. Data from the 2007 to 2011 time period, also referred to as *prediction period*, is used to assign ground truth labels, *i.e.* whether or not two researchers, for whom features are computed, collaborate. Pairs of authors (identified by bubbles around them) that did not collaborate during the history period are selected. Some will collaborate in the prediction period (denoted by addition of dashed edges in prediction period graph), and some of them will not. These pairs of nodes are divided into two node disjoint sets, *i.e.* a training set (black bubbles) and a test set (red / light gray bubbles). Depending on the line type used for each bubble in Fig. 2, every pair of authors in the training and test set will either be a positive example of collaboration (solid line bubble), or no collaboration (broken line bubble). Disjointness in space between the two sets is achieved by ensuring that no node that appears in the training set may appear in the test set (and vice versa). In Fig. 2 that can be verified visually by the non-overlap of any black bubbles with red bubbles.

5.2 Feature Extraction

Using the parent graph and its various projections, we compute properties of author nodes. These properties, labeled $P1$ through $P13$, are listed in the first column of Table 2. Since the classifier will operate on features of a pair of authors, the properties of two authors for which future collaborations being predicted need to be combined into a single feature. We use various functions to obtain a feature from two nodes' properties. These functions, labeled $F1$ through $F9$, are listed as follows.

F1 Sum

F2 Difference

F3 Product

F4 Difference normalized by sum

F5 Intersection set cardinality[2]

F6 Average

F7 Maximum

F8 Author 1

F9 Author 2

In Table 2, the intersections of node properties and functions that contain checkmarks are property-function combinations that have been considered as predictors.

5.3 Feature Selection

Following the generation of the features listed above, they are ranked in order of their usefulness for predicting collaboration. Three different feature ranking methods were used. We used the WEKA [11] platform's implementation of these ranking metrics.

5.3.1 Information Gain

The information gain attribute evaluator measures the information gain of each attribute with respect to the class. This can be explained in terms of the information theoretic measure, mutual information [8]. In the information theory community, information gain is also known as mutual information. The mutual information $I(X_i; C)$ of two random variables X_i and C modeling features, is defined as

$$I(X_i; C) = H(X_i) - H(X_i|C), \tag{1}$$

Here, $H(X_i)$ is the entropy of X_i, and $H(X_i|C)$ is the conditional entropy of X_i conditioned on C. Then the information gain $IG(X_i)$ of a feature X_i with respect to class label C is defined as

$$IG(X_i) = \sum_i I(X_i; C) \tag{2}$$

[2] The intersection set cardinality is computed for the sets of elements in properties P10, P11, P12 and P13.

5.3.2 Gain Ratio

The gain ratio of a feature (modeled by a random variable X_i) with respect to the ground truth label (modeled by another random variable C) is their mutual information normalized by the entropy of X_i. Mathematically, this is defined as;

$$GR(X_i, C) = \frac{I(X_i; C)}{H(X_i)}. \tag{3}$$

5.3.3 Chi-Squared Statistic

The χ^2-statistic is the sum of squared error divided by the target value. Mathematically, it is defined as follows.

$$\chi^2 = \frac{\sum(Observed - Expected)^2}{Expected} \tag{4}$$

The χ^2-statistic is used to examine if the distributions of categorized variables differentiate from each other.

5.4 Classifier Design

We evaluated three different supervised learning algorithms; Naïve Bayes (NB), C4.5 Decision tree (simply referred to as C4.5 from here on) and Support Vector Machine (SVM). For NB and C4.5 we used the Naïve Bayes and J48 Java implementations available in WEKA [11]. For SVM we used the libSVM library implementation [6] and WEKA [11].

5.5 Performance Evaluation

A modified adaptation of N-fold cross validation is used for evaluating all considered classifiers. Due to the particular way training set D_{trg} and test set D_{test} were prepared it is not possible to use N-fold cross validation in its basic form. In the adapted form of N-fold cross validation we apply N-part partitions to both the test and training sets such that;

$$D_{trg} = \bigcup_{i=1}^{N} D_{trg}^{(i)} \text{ , and } D_{test} = \bigcup_{i=1}^{N} D_{test}^{(i)}, \tag{5}$$

where for any $1 \leq i, j \leq N$ and $i \neq j$,

$$D_{trg}^{(i)} \bigcap D_{trg}^{(j)} = \Phi \text{ , and } D_{test}^{(i)} \bigcap D_{test}^{(j)} = \Phi. \tag{6}$$

For the i-th validation, $D_{test}^{(i)}$ and $D_{trg}^{(i)}$ are both withheld from the test and training data sets, respectively. This adaptation N-fold cross validation retains the benefits of classic N-fold cross validation while maintaining spatial disjointedness of training and test data sets.

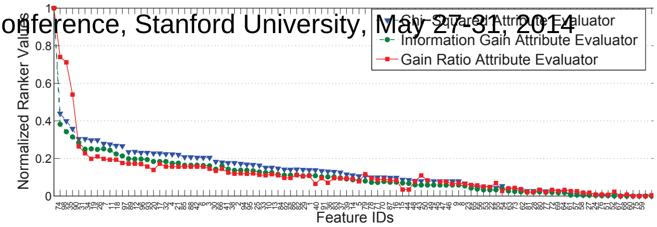

Figure 3: Normalized and averaged 10-fold ranking metrics of all 98 features, reverse sorted by median rank.

6 Results

6.1 Feature Ranking Results

In Fig. 3 features are shown ranked by their median rank across the three rank metric vectors.[3] The generally decreasing trend and only minor and infrequent upticks in all three lines show that all ranking metrics order features similarly. The design of Machine Learning classifiers is not linear, but an iterative process. While the feature ranking metrics order features by their contribution to classification, performance results plotted as a function of the number of features provides more guidance for selecting a cutoff for the number of features to use.

To determine the number of features to use, we plotted the True Positive (TP) rate, False Positive (FP) rate, precision, and area under ROC curve. We plotted these quantities for all three classifiers we considered in the subsequent sections; Naïve Bayes (NB), C4.5 decision tree, and Support Vector Machine (SVM) in Fig. 4. All numbers in this plot are computed over 10-fold cross validation. Lines for the same classifier share the same line color and marker, while lines for the same classification metric share the same linetype (refer to provided legend). Ideally, TP rate, precision, and ROC should be 1 while the FP rate should be 0. Results were plotted for up to 12 features. As the plot shows, all lines become quite stable when only three features are used. However, there is a notable increase in the ROC of the NB classifier when the eighth feature is added. but remain stable thereafter in the plotted range. Very notably, when the 10^{th} feature is added, the SVM's TP rate, and ROC area drop significantly with a simultaneous jump in its FP rate. We explain the sudden deterioration in SVM's performance by overtraining. Based on this plot we selected a cutoff point of eight features

[3] Note that features are referred to either by a Linear Feature ID (FID) which is an integer in the range 1 to 98, or the row-column IDs in Table 2. The mapping of these two IDs for the first 12 most significant features is given in Table 3.

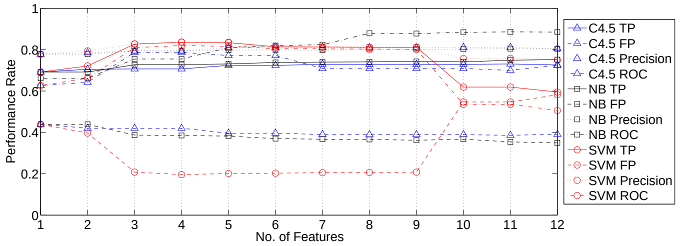

Figure 4: TP rate, FP rate, precision and ROC area of NB, C4.5 decision tree and SVM classifiers plotted as a function of number of features used.

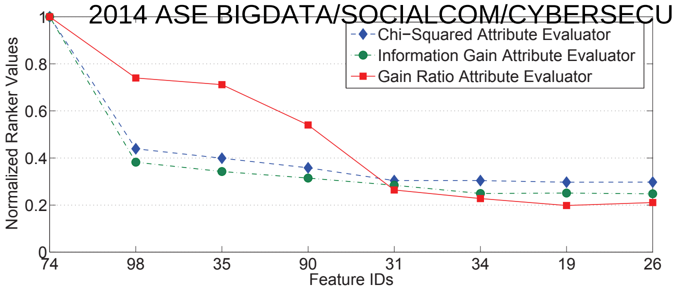

Figure 5: Normalized and averaged 10-fold ranking metrics of 8 most significant features, reverse sorted by median rank.

IG	GR	χ^2	Med. Rank	Lin. FID	Feature Description
74	74	74	1	74	[P10-F5] Common Words in Title
98	98	98	2	98	[P13-F5] Common References
35	90	35	3	35	[P5-F3] Product of Triangles
90	66	90	4	90	[P12-F5] Common Co-authors
19	82	31	5.5	19	[P3-F4] Diff / Sum of Closeness Centrality
31	19	34	6	31	[P5-F1] Sum of Triangles
34	70	7	7	34	[P5-F6] Avg of Triangles
26	18	19	8.5	26	[P4-F4] Diff / Sum of Avg Neighbor Deg
18	78	26	9	18	[P3-F2] Diff of Closeness Centrality
7	79	11	10	7	[P1-F3] P.A. Degree
11	71	18	11	11	[P2-F2] Diff. Squ. Clustering
97	21	97	12	97	[P13-F3] P.A. References

Table 3: [Columns from left to right] Top-12 most significant features ranked by IG, GR and χ^2 , their median rank, the linear feature ID (FID) and their descriptive names.

for the purpose of prediction. As the Fig. 5 (a zoomed in view of Fig. 3) shows, in all three ranking metrics, the eighth feature provides $20 - 25\%$ of information relative to the most significant feature.

Table 3 tabulates the top eight features, as well as the next four features, ordered by all three ranking metrics, and their median rank. Note that all features (F1 through F7) in Table 2 are based on the properties of both pairs of authors, not just any single author (F8 and F9).

Based on Fig. 4, most performance metrics become quite stable after including just three features for classification (exception being SVM's performance when 10^{th} feature is included, as described above). It can be argued that significantly fewer features can be used to similar effect. Certain applications may choose to trade the bump in TP rate of the NB classifier when eight features are included for a significantly simpler classifier that uses only three features. Nevertheless, there being no such constraint on us, in the following subsection we proceed with using eight features for all considered classifiers. It should be noted that the three most useful features include semantic features (1. number of common words in paper titles, 2. number of common cited references) as well as a topological feature (3. preferential attachment or the product of triangles).

TP rate	FP rate	Precision	F-Measure	Class
0.39	0.01	0.95	0.55	1
0.99	0.61	0.67	0.82	0
0.74	0.37	0.80	0.71	Wt.Avg.

Table 4: Classification results of Naïve Bayes classifier.

TP rate	FP rate	Precision	F-Measure	Class
0.38	0.00	0.99	0.54	1
0.99	0.62	0.70	0.82	0
0.74	0.37	0.82	0.71	Wt.Avg.

Table 5: Classification results of C4.5 decision tree classifier.

It should be noted that 10-fold cross validation results for all classifiers discussed below were also generated using only the top three features. The results were very similar to the results obtained using eight features that are described, and discussed in the following subsections.

6.2 Classification Results

For each classifier we report confusion matrix as well as TP rate, FP rate, precision, F-measure, and ROC Area (referred to as 'AUC' in [24]). While Wang *et al.* [24] reported their predictor performance in terms of ROC area as well, readers are cautioned that Wang *et al.* reported their performance for only the top 500 most certain predictions of future collaborations. It should also be kept in mind that Wang *et al.* did not ensure spatially disjoint training and test data sets.

Here we describe the results of adapted 10-fold cross validation for all three considered classifiers. The cardinality of the validation set is $|D_{test}| = 13,900$, consisting of $5,700$ instances of collaborations labeled $C = 1$ and $8,200$ instances labeled $C = 0$. For the first of the 10 validation runs, $D^{(1)}_{trg}$ and $D^{(1)}_{test}$ are withheld from training and test data sets. Similarly, for the second run $D^{(2)}_{trg}$ and $D^{(2)}_{test}$ are withheld from training and test data sets, and so on.

Table 4 contains TP rates, FP rates, precision, and F-measure for the two classes (first two data rows), as well as their weighted average (last row). ROC area for NB classifier is 0.88. The accuracy of the NB classifier using eight features is 74%, while for three features it is slightly lower at 73%.

Table 5 contains TP rates, FP rates, precision, recall, F-measure, and ROC area for the two classes (first two data rows), as well as their weighted average (last row). ROC area for C4.5 decision tree is 0.61. The accuracy of the C4.5 decision tree classifier using eight features is 74%, while for three features it is slightly lower at 71%.

Table 6 contains TP rates, FP rates, precision, recall, F-measure, and ROC area for the two classes (first two data rows), as well as their weighted average (last row). ROC area for SVM classifier is 0.80. When we use eight features the SVM classifier has an accuracy of 81%, the highest of all classifiers, and for

TP rate	FP rate	Precision	F-Measure	Class
0.76	0.15	0.77	0.76	1
0.85	0.24	0.83	0.84	0
0.81	0.21	0.81	0.81	Wt.Avg.

Table 6: Classification results of SVM classifier.

Class	Naïve Bayes		C4.5 dec. tree		SVM classifier	
	$\widehat{C}=1$	$\widehat{C}=0$	$\widehat{C}=1$	$\widehat{C}=0$	$\widehat{C}=1$	$\widehat{C}=0$
$C=1$	**1,977.3**	3,152.7	**1,928.9**	3,201.1	**3,882.3**	1,247.7
$C=0$	108.1	**7,270.9**	27.0	**7,352.0**	1,137.9	**6,241.1**

Table 7: Confusion matrix of Naïve Bayes, C4.5 decision tree, and SVM classifier.

three features a slightly higher accuracy of 83%.

Table 7 is the confusion matrix[4] of the Naïve Bayes, C4.5 decision tree, and SVM classifier. The predicted class of a pair of authors is denoted by $\widehat{C}$.

6.3 Discussion

6.3.1 Features

Table 3 tabulates the top 12 features for the purpose of prediction, ordered by median of ranks of all three feature ranking algorithms. The most important observation to be made is that feature P10-F5, the number of common words in paper titles, and feature P12-F5 (number of common referenced papers) were identified as the first and second most significant features by all three feature rankers. This is significant because most lower ranked features are A) ones that earlier works have repeatedly established as significant, and B) consist almost exclusively of topological features. Our findings show that high degree of overlap in previous publications is an even more significant predictor of new collaborations than the proximity of two researchers in collaboration network. Our principal motivation for using semantic features is that most earlier work did not explore them extensively.

Al-Hasan *et al.* [2] used a combination of the author proximity features, and topological features and identified number of common keywords, a semantic feature, as the most significant feature of all. None of their feature performed better than the number of common title words and the number of common references. Fire *et al.* [9] introduced a new subset of dyadic features, called *friends-features*, which included features based on vertex degrees, number of common friends, and preferential attachment (PA) score. Tylenda *et al.* [23] used clustering coefficient and distance between collaborating pairs. We can see that the performance of clustering coefficient is not optimal and none of its features lies in the top 12.

6.3.2 Classifiers

Of the three classifiers that are evaluated for prediction, the SVM classifier performs best in terms of TP rate, FP rate, recall, and F-measure when averaged over both classes. All three classifiers have very similar precision between $80-82\%$. However, both C4.5 decision tree, and SVM classifiers boast higher precision than the Naïve Bayes classifier. NB outperforms SVM's area under ROC curve by a relative margin of 10%, while C4.5 decision tree comes in last. However, a closer look at the classwise classification performance reveals that NB offers a very low TP rate of just 38% for collaborating pairs of authors compared to SVM's 75%. This is however, offset by a higher FP rate of SVM for collaborators. Similarly, the TP rates of non-collaborating authors for both NB and C4.5 decision tree is much higher than SVM's, but both also have very high FP rates of 61% compared to SVM's lower FP rate of 24%. There is no one classifier that clearly outperforms all others by all performance measures. Generally, the choice of which classifier should be employed will also depend on the type of application, and the costs of misclassification of instances belonging to each class. However, assuming that a higher value is attached with the correct prediction of future collaborators than non-collaborators, and assuming that a 15% FP rate for collaborators is not too high a cost, we judge SVM to be the best classifier among the three considered here. Clearly, the C4.5 decision tree is outperformed or matched in almost any measure by NB. Its ROC area is less than 62% which makes it only slightly better than a random predictor. Like NB, it has a low TP rate of $37-38\%$ for collaborating authors, and a very high FP rate of $61-62\%$ for non-collaborating authors. It is pertinent to mention that machine error can change the accuracy of collaboration prediction in real life depending upon the hidden factors such as geography, cultural backgrounds, affiliations, and past experiences etc.

6.4 Performance Comparison with Prior Approaches

Al Hasan *et al.* [2] and Liben-Nowell and Kleinberg [18] used random predictors as their baselines (Al Hasan *et al.* used a uniform Bernoulli random predictor, Liben-Nowell and Kleinberg non-uniform). For the IEEE and ACM dataset we used, the same baseline approach used by Al Hasan *et al.* and Liben-Nowel and Kleinberg gives an accuracy of less than 1%. Al Hasan *et al.* reported a high accuracy rate for a list of eight classifiers ranging from $81-90\%$ for both classifiers using nine features for the BIOBASE data set, and four features for the DBLP data set. However, it should be noted that Al Hasan *et al.*'s approach did not ensure spatial disjointness between training, and test data sets. We tested Al Hasan *et al.*'s features on the IEEE-ACM data set. We employed four of the features used by Al Hasan *et al.* [2] available to us with NB, C4.5 decision tree, and SVM classifiers. The NB, C4.5 decision tree, and SVM classifiers had accuracies of 65%, 74% and 71%, respectively. Full results for SVM have been shown in Table 8. However, clearly our SVM classifier outperforms Al Hasan *et al.*'s on almost all measures. Liben-Nowell and Kleinberg analyzed each feature separately and reached a highest accuracy of only 16%.

7 Conclusions

In this paper we explore three different link predictors based on supervised machine learning algorithms. A significant contribution of our work lies in the collection of a very large new data set spanning 10 years of publications in all flagship conferences of 38 IEEE Societies, and 32 ACM SIGs. The data set collected is very rich in details such as paper abstracts and citation information. This enabled us to explore many more semantic features

TP rate	FP rate	Precision	F-Measure	Class
0.43	0.10	0.75	0.55	1
0.90	0.57	0.70	0.78	0
0.71	0.38	0.72	0.69	Wt.Avg.

Table 8: Classification results of Al Hasan *et al.* using SVM classifier.

[4] Fractional values result from the averaging over 10-folds used for validation.

than prior works. Most significantly perhaps, the number of common title words in prior publications of two authors, as well as the number of common referenced papers between them were determined to be the two most significant features. This suggests that future collaborators may evaluate each other based on the degree to which their prior works have narrowly converged on the similar research problems. Moreover, our approach distinguishes itself from all prior approaches in that it ensured complete spatial separation between training and test data sets. Prior approaches did not maintain this clear separation. We formulated the link prediction problem as a binary classification problem. We explored, and ranked 98 different features in order of their usefulness and identified the eight most useful ones. Using these eight features we evaluated three different classifiers, of which the SVM classifier performed best overall for most anticipated applications of such a predictor. The permanent separation between instances of the training and test data necessitated an adaptation to the classic N-fold validation technique.

References

[1] L. A. Adamic and E. Adar. Friends and neighbors on the web. *Social networks*, 25(3):211–230, 2003.

[2] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM'06: Workshop on Link Analysis, Counter-terrorism and Security*, 2006.

[3] N. Benchettara, R. Kanawati, and C. Rouveirol. Supervised machine learning applied to link prediction in bipartite social networks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*, pages 326–330. IEEE, 2010.

[4] S. Bird, E. Klein, and E. Loper. *Natural language processing with Python*. O'Reilly Media, Inc., 2009.

[5] J. E. Celis, P. Gromov, M. Østergaard, P. Madsen, B. Honoré, K. Dejgaard, E. Olsen, H. Vorum, D. B. Kristensen, I. Gromova, et al. Human 2-d page databases for proteome analysis in health and disease: http://biobase.dk/cgi-bin/celis. *FEBS letters*, 398(2):129–134, 1996.

[6] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[7] A. Clauset, C. Moore, and M. Newman. Hierarchical structure and the prediction of missing links in networks. In *Nature*, pages 98–101, 2008.

[8] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[9] M. Fire, L. Tenenboim, O. Lesser, R. Puzis, L. Rokach, and Y. Elovici. Link prediction in social networks using computationally efficient topological features. In *IEEE third international conference on Privacy, security, risk and trust (passat), and IEEE International Conference on Social Computing*, 2011.

[10] A. Hagberg, P. Swart, and D. S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), 2008.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: An update. *SIGKDD Explorations, Volume*, 11, 2009. Issue 1.

[12] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[13] V. E. Krebs. Mapping networks of terrorist cells. *Connections*, 24(3):43–52, 2002.

[14] J. B. Lee and H. Adorna. Link prediction in a modified heterogeneous bibliographic network. In *Advances in Social Networks Analysis and Mining (ASONAM), 2012 IEEE/ACM International Conference on*, pages 442–449. IEEE, 2012.

[15] J. Leskovec. Stanford network analysis package (snap). *URL http://snap. stanford. edu.*

[16] J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins. Microscopic evolution of social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 462–470. ACM, 2008.

[17] M. Ley. The dblp computer science bibliography: Evolution, research issues, perspectives. In *String Processing and Information Retrieval*, pages 1–10. Springer, 2002.

[18] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[19] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 243–252. ACM, 2010.

[20] D. R. Radev, P. Muthukrishnan, V. Qazvinian, and A. Abu-Jbara. The acl anthology network corpus. *Language Resources and Evaluation*, 47(4):919–944, 2013.

[21] G. Salton and M. J. McGill. Introduction to modern information retrieval. 1983.

[22] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han. Co-author relationship prediction in heterogeneous bibliographic networks. In *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on*, pages 121–128. IEEE, 2011.

[23] T. Tylenda, R. Angelova, and S. Bedathur. Towards time-aware link prediction in evolving social networks. In *Proceedings of the 3rd Workshop on Social Network Mining and Analysis*, page 9. ACM, 2009.

[24] C. Wang, V. Satuluri, and S. Parthasarathy. Local probabilistic models for link prediction. In *Seventh IEEE International Conference on Data Mining ICDM 2007*, pages 322–331. IEEE, 2007.