Hindawi
Complexity
Volume 2018, Article ID 9104387, 12 pages
https://doi.org/10.1155/2018/9104387

Research Article

Predicting Hidden Links in Supply Networks

A. Brintrup ,[1] P. Wichmann,[1] P. Woodall,[1] D. McFarlane,[1] E. Nicks,[2] and W. Krechel[2]

[1] *Institute for Manufacturing, Department of Engineering, University of Cambridge, Cambridge CB3 0JB, UK*
[2] *The Boeing Company, Seattle, WA, USA*

Correspondence should be addressed to A. Brintrup; ab702@cam.ac.uk

Received 2 May 2017; Revised 17 November 2017; Accepted 20 December 2017; Published 30 January 2018

Academic Editor: Pietro De Lellis

Manufacturing companies often lack visibility of the procurement interdependencies between the suppliers within their supply network. However, knowledge of these interdependencies is useful to plan for potential operational disruptions. In this paper, we develop the Supply Network Link Predictor (SNLP) method to infer supplier interdependencies using the manufacturer's incomplete knowledge of the network. SNLP uses topological data to extract relational features from the known network to train a classifier for predicting potential links. Using a test case from the automotive industry, four features are extracted: (i) number of existing supplier links, (ii) overlaps between supplier product portfolios, (iii) product outsourcing associations, and (iv) likelihood of buyers purchasing from two suppliers together. Naïve Bayes and Logistic Regression are then employed to predict whether these features can help predict interdependencies between two suppliers. Our results show that these features can indeed be used to predict interdependencies in the network and that predictive accuracy is maximised by (i) and (iii). The findings give rise to the exciting possibility of using data analytics for improving supply chain visibility. We then proceed to discuss to what extent such approaches can be adopted and their limitations, highlighting next steps for future work in this area.

1. Introduction

Supply networks emerge as manufacturing firms become dependent on procuring subcomponents or services from other firms in order to produce their own products. Outsourcing aspects of production to suppliers allows manufacturers to specialise and scale-up by setting up dedicated production lines. However, the practice exposes companies to risk of disruptions on material flow along the emergent network of dependencies. As supply networks grow in their size and complexity, it becomes increasingly difficult for companies to keep track of who is in their network and thus what risks they are exposed to. Manufacturers are typically aware of only their first tier suppliers, gradually losing visibility over their extended network.

Knowing the extended network is important so that appropriate risk mitigation plans can be prepared in advance, to ensure that production continues smoothly should a disruption happen. For example, in the aerospace sector, 65%–80% of the final cost of aerospace production is dedicated to suppliers, while majority of delays and quality issues can also be traced back to supply chain issues [1].

Although several solutions have been proposed for the identification of interdependencies (hereafter links) between suppliers, most of these require the willingness and ability of suppliers to share information. Among these, the most commonly practiced method for a manufacturer is the questioning of first tier suppliers on their procurement partners. Powerful manufacturers could enforce a contractual auditing process, which includes supplier interviews, performance monitoring, and collection of data during visits. Some manufacturers ask their suppliers to select their own suppliers from an approved supplier list. However, it is well documented that suppliers do not have sufficient incentives to share information about their own suppliers, as this could lead to the buyer negotiating directly with their suppliers to reduce margins.

Manufacturers increasingly subscribe to third party companies that independently gather supplier information by surveying suppliers. These companies present economies of scale advantage; as many manufacturers share common suppliers, they can reuse data and cross map industrial ecosystems as more companies sign up to the service. However, this approach presents similar challenges as suppliers might not want to share private data. Third party databases also present an asymmetry of information challenge, as manufacturers have no way of verifying the information provided by suppliers.

Another strategy for supply network visibility is to deploy traceability technology such as RFID tagging. These methods increase visibility over specific stages of the production process; however, they require adoption by the rest of the supply chain, a process over which the manufacturer would not have control.

As supply network visibility gains urgency, there is a need for complimentary methods that do not rely on suppliers' willingness or ability to share information.

In this paper, we present an alternative approach that combines information from the topology of the existing network with relational information that is obtained from the topology, in order to automatically infer links between suppliers. The link inference is then made by using classification algorithms from the field of machine learning. The approach has been tested with three empirical samples from the global automotive network: namely, the supply networks of Saab, Volvo, and Jaguar Land Rover. Our results show that the combination of topology and relationship extraction can indeed be used to predict interdependencies in the case study networks, giving rise to the exciting possibility of using data analytics for improving supply chain visibility.

Our contribution thus includes a method for detecting invisible dependencies in supply networks that does not rely on suppliers to share information and three case studies from the automotive industry that illustrate how the approach can be adopted.

This paper is organized as follows. In Section 2, we discuss related work and contributions. In Section 3, we will discuss how supply chain visibility can be characterised as a link prediction problem and we develop a method that combines supplier attributes with topological information. Section 4 presents and discusses experimental results. Section 5 concludes the paper by summarizing findings and limitations and identifying future avenues of research.

2. Literature Review

A supply network involves manufacturers buying products from one another to produce their own products. Consider a supply network as a graph $G\{N, L\}$, where suppliers are represented by a nodeset N, and procurement relation between suppliers is represented by a linkset L. Links are directed, depicting the direction of material flow from one supplier to another. The direction of the link determines whether a supplier is acting as a buyer or a seller in a particular relationship instance.

Given such a directed graph, all the possible links in the graph are of size $N(N-1)/2$. Link prediction is defined as the estimation of the likelihood that two nodes interact with each other, based on the observed network structure [2]. In other words, we need to distinguish between links that exist and links that do not exist. The problem could be viewed as a binary classification problem, where links are classified into positive (existing links) and negative (nonexisting) links.

Machine learning algorithms were developed to study large datasets in order to identify patterns and make predictions. The learning algorithm labels each possible edge of the network as "exists" or "does not exist" in the network, according to the feature vector associated with the link. A training set of known samples is used to teach the algorithm, which is then applied to new samples to predict unknown instances. Several algorithms exist to solve classification problems (please refer to [3] for a review).

Although link prediction problems seem to be a natural fit to the application of machine learning, these have thus far been inadequate for various reasons [4]. First reason is that the number of features available for each link instance has been limited to topological features. Furthermore, when machine learning on link prediction is deployed with flat data representations, relational features are ignored, which could contain a wealth of additional data. Using only topological features is also problematic because most classification algorithms assume that the data sample is independent and identically distributed, whereas in a network they are not; networks by their nature consist of heterogeneous distribution of topological features. Second, using topology only could give imprecise results irrespective of the features associated with nodes. Third, the computational cost of calculating the features of all possible instances for training an algorithm is high.

Researchers have thus devised alternative methods for predicting links in networks, based on node characteristics as well as topological network information, giving rise to the field of graph mining. In graph mining, one assumes that the information is represented through relations; therefore, knowledge emerges from the interaction of nodes through links. While the goal of machine learning based classification is to distinguish between classes with the highest possible precision and recall, the goal of graph mining is not only accuracy of prediction but also the understanding of the behaviour of connectivity and using that knowledge for predictive analytics.

Most work in the graph mining field has largely been motivated by the need for analyzing entity relationships in social networks, and other digitised large datasets such as product recommender systems. Three main categories of approaches prevail (Lü and Zhou, 2011): similarity based algorithms, maximum likelihood algorithms, and probabilistic models.

Similarity based algorithms work by computing a similarity score between each pair of nodes in a network based on their relations. Then, by comparing the similarity scores of all links, one obtains the likelihood with which a link between a given pair of nodes exists. Similarity scores can be based on node characteristics or topological relationship patterns.

Some of the most well-known algorithms in this category are common neighbours (CN), Jaccard coefficient [5], Katz [6], and Leicht-Holme-Newman Index (LHNI) [7], path-based similarity algorithm [8], and the more recent Bayesian estimation algorithm proposed by [9]. For example, CN simply calculates the similarity between nodes by counting the number of their common neighbours. The likelihood that two nodes are connected increases with the number of common neighbours they have. The LHNI quantifies similarity by assuming that nodes are similar if their immediate neighbours are similar.

A fundamental assumption of these algorithms is that the more similar the two nodes are, the more likely they are to share a link. While this assumption has strong foundations in social science, it is hard to justify in a supply network where companies with similar characteristics would often be those that compete with one another, rather than engaging in a buyer supplier relationship.

Maximum likelihood algorithms start by assuming a predefined topology that the network is most likely to have, including the existence of a hierarchy, large hubs, or a set of communities. After this assumption, a model is built to calculate the likelihood of potential links that are not found in the original graph. Within this category, Clauset et al. [10] proposed the hierarchical random graph model, which builds a dendrogram representing a hierarchical abstraction of the network under study. A set of connection probabilities is inferred from a "consensus dendrogram" that would most accurately represent the hierarchical network, which is then used to predict likelihood of links preserving the hierarchical structure. These algorithms offer important lessons on how fundamental properties that drive a given network structure impact the likelihood with which new nodes that enter the network will be connected to existing nodes in the network.

However, empirical data on supply network topologies is sparse in the literature. Four previous studies exist: Thadakamalla et al. [11] and Hearnshaw and Wilson [12] have built models on scale-free structures, and Brintrup et al. [13, 14] mapped the global automotive network and the Airbus supply network using empirical data. The global automotive network displayed an exponential degree distribution; whereas the Airbus network had too small a sample size to determine significant patterns in scale. The scarcity of empirical examples and their conflicting results prevent us from opting for methods that depend on a priori assumptions on topology.

The third approach includes probabilistic models, which optimise a network topology according to datasets, giving the probability of a new link as conditional to the estimated parameters (e.g., [15, 16], Getoor et al., 2001). It has been noted that these approaches could be computationally intensive as inference is done by creating the complete ground network, which limits their scalability (Lü and Zhou, 2011).

More recently, hybrid approaches have been deployed that bring together both attribute data on nodes and structural information, which is useful to bring domain knowledge into the prediction effort where neither individual features nor topological features are dominant factors. Examples of this approach include Aggarwal et al. [17] and Al-Hassan et al. [18]'s work on terrorism networks. For example, in [17], the link prediction process involves first a macroprocessing step to extract clusters and a second microprocessing step to find new links on those clusters based on node properties. These methods seem to offer promising accuracy but are inevitably domain specific and thus need knowledge of the network under study.

In a similar vein to hybrid approaches, we propose a combined approach specific to supply networks, which is described in the next section.

3. Characterising Supply Network Dependencies: A Link Prediction Problem

Our inquiry is about estimating the likelihood that two suppliers interact with each other, based on an incomplete observation of the supply network. For gathering such an estimation, we propose the Supply Network Link Predictor (SNLP) method, which is composed of the following:

(1) Data that would form the incomplete observation

(2) The features extracted from the data that can inform the relevant relational patterns for use in estimating the likelihood

(3) A method for relating extracted features to the estimation

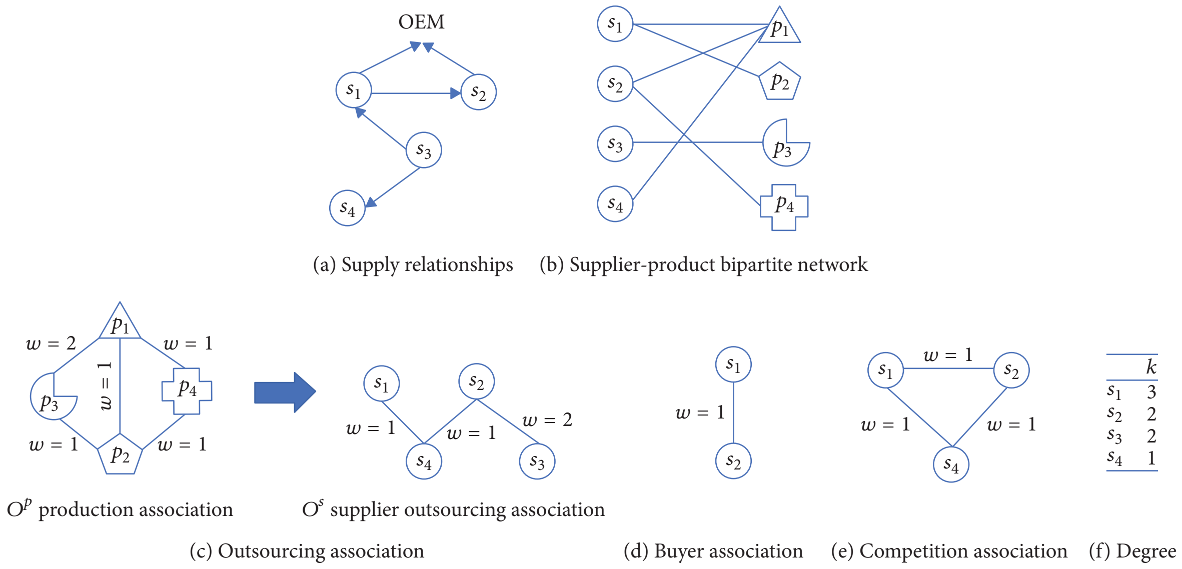
For (1), our starting point is the minimum amount of data a manufacturer can have over its supply network. This data includes the identity of suppliers that are known to be attached to the network (the nodes); known links between suppliers (the links); and known products that each supplier produces (the product distribution over the network) (Figure 1(a)).

We shall deliberately ignore dynamical data relating to material flow, as these can change frequently and might not be uniformly available to the manufacturer across the known network. Obviously, the buyer might be acting as buyer in some instances and seller in other instances. Hence, nodes can have multiple roles. To avoid confusion, the terms buyers and sellers are used to refer to a particular relationship instance between two nodes hereon.

Next is identification of features that could be extracted from the data that would point to the likelihood of a link. Using the basic network, we decided on three types of relational patterns that might be extracted: outsourcing associations; competitive associations; and buying associations. In addition, we extract a topological feature, that of degree of suppliers. These are described next.

(1) Outsourcing Association. In a supply network, products are transmitted on the links between buyers and sellers. Each buyer decides on what to produce and what to outsource. Hence, the buyer uses the outsourced product in order to produce its own product. For instance, a buyer might buy fabric to produce car seats. If we know the products that have such production dependencies and which supplier produces which products, then we can use this information to predict potential links between suppliers. To do so we use the following procedure.

In the original network (e.g., Figures 1(a) and 1(b)), each supplier produces a number of products and links to a buyer, which also produces a number of products. Each of these products might potentially be transmitted on the links; however, we do not know which. Furthermore, the buyer might be using the products of its supplier for producing any



(a) Supply relationships (b) Supplier-product bipartite network

O^p production association O^s supplier outsourcing association

(c) Outsourcing association (d) Buyer association (e) Competition association (f) Degree

Figure 1: An example supply network (a), and products supplied by each supplier (b) followed by the relational (c, d, e) and topological (f) features extracted from it.

of its own products and we do not know which. Therefore, there can be a potential dependency between each product of the buyer and each product of the seller.

We can thus create a potential outsourcing association between each product category sold by the supplier and each product category produced by the buyer. The outsourcing association O_{p_i,p_j} between each product can be represented in the form of a network $O^p\{N, L\}$ (Figure 1(c)) between each unique product category $p_i \in N$, where the set of N nodes represent products and the set of L^p represents an outsourcing association between them. Each link representing a potential outsourcing dependency has a weight $w(O_{p_i,p_j})$ equal to the number of times the associated products occur in a buyer's portfolio and a seller's portfolio in the network. By adding weights to links one can refine their interpretation. The greater the weight on a given associative link, the higher the likelihood of an actual dependency between the nodes.

After creating the association network between products, we must extrapolate this information to associations between suppliers that produce a set of products. Our assumption here is that the more cross-associative products two suppliers produce, the more likely they are to share a link. For example, if a buyer is producing car seats and a seller is producing fabric, as well as sponge for car seats, then the two might be more likely to share a link as there is more dependency between them. Given the transaction cost of setting up a link between buyers and sellers and the synergies between logistic planning for two products impacting the buyer's production schedule, the buyer might be inclined to buy a bundle of goods from the supplier rather than from two separate suppliers. Of course, there would be a number of opposing factors, such as the risk of overdependence to a supplier and economies of scale from specialising in a smaller portfolio of products that might result in the opposite effect; however, our purpose is not to test these hypotheses but to simply use the potential likelihood as an information source to predict hidden relationships.

To associate between two suppliers, we create a supplier outsourcing association network $O^s\{N, L\}$, where each node is a supplier and each link is an outsourcing association that represents the sum of product associations between each supplier's product portfolios. To do so, each supplier node's product portfolio is compared with one another, and the product association weight between each product in the product association network is added to the supplier association link. Hence, every pair of suppliers in the network will have an outsourcing association weight of (O_{s_i,s_j}). If there is no link between s_i, s_j in O^s, then $w(O_{s_i,s_j}) = 0$.

(2) Buyer Association (B). The second relational pattern we extract is buyer association. The goods that buyers purchase from sellers might have dependencies on each other: for example, the product of one seller might be compatible with another seller's product. In this case, a buyer that needs both products will need to buy from these two sellers. Over time this can give rise to a "buyer association" between sellers. Hence, a firm connecting to a supplier would also be likely to connect to the other supplier that the supplier has a buyer association to. This information can be useful in predicting missing links from the network; an OEM might know that its first tier supplier is connected to one tier 2 supplier and can deduce that tier 1 is connected also to another supplier that is not visible to it.

We create a buyer association network (Figure 1(d)) to extract association information between sellers. In the network $B\{N, L\}$, a node represents a supplier and link represents a buyer association between them. To create the network, the original network is parsed, where each supplier is associated with another supplier if they sell to the same buyer. The weight on the link is increased each time the two suppliers sell to the same buyer. The higher the weight on the link, the higher the buyer association between these two suppliers.

Hence, every pair of suppliers in the network will have a buyer association weight of $w(B_{s_i,s_j})$. If there is no link between s_i, s_j in B_{s_i,s_j}, then $w(B_{s_i,s_j}) = 0$.

(3) Competition Association (C). In this association type, we search for competitive relations between suppliers. Our working hypothesis is that the more overlapping products there are in the portfolios of two suppliers, the more likely they are to be competitors and thus less likely to share dependency links. In addition, when two firms produce the same products, they are less likely to become dependent on each other. However, there are a number of opposing lines of thought. When suppliers face capacity constraints, they might work together to increase economies of scale and pool resources to sell to a buyer together. In addition, two suppliers might be producing the same product category but slightly different models within that category. In this case, they might supply subcomponents to one another and do not compete as they have segmented their market. Regardless of the direction of the relationship between competition association and the existence of a dependency between suppliers, the competition perspective might prove informative in the prediction of links. We create a competition association network (Figure 1(e)) $C\{N, L\}$, where the nodes represent suppliers and links represent competition association.

We do so by comparing the product portfolios of each supplier in the original network. If two suppliers produce the same product, a link is created between them. The weight on the link is equal to the number of overlapping products.

Hence, every pair of suppliers in the network will have a competition association weight of $w(C_{s_i,s_j})$. If there is no link between s_i, s_j in C_{s_i,s_j}, then $w(C_{s_i,s_j}) = 0$.

(4) Degree (D). Finally, the degree information of suppliers in the original network is taken into account, where degree D is the number of links a supplier node has (i.e., incoming and outgoing links) (Figure 1(f)). Previous studies have shown that supply networks have hub firms ([13], Kito et al. 2016, Thadakamalla et al. 2014). These hub firms have a large number of links compared to other nodes in the network. Additionally, the likelihood for a supplier to acquire more links in the network increases with the number of links it already has. Although the exact scale of this property is debated with some studies showing an exponential degree distribution ([13], Kito et al. 2016) and some scale-free property [11, 12], the existence of "hub" firms is well established in the ongoing debate. Hence, we opt to make use of this topological knowledge by introducing the degree of a node as a feature that might be useful in predicting the likelihood of missing links attached to it.

Hence, the features that are identified to inform relevant relational patterns for estimating the likelihood of two nodes interacting include $w(O_{s_i,s_j})$, $w(B_{s_i,s_j})$, $w(C_{s_i,s_j})$, and D_{S_i}. The dependent variable is L_{s_i,s_j}.

The final step is the definition of a method for relating extracted features to the estimation. We opt for the use of classification algorithms from the field of machine learning.

The classifier algorithm assigns each possible additional edge of the network either to the positive class, that is, "exists," or to the negative class, that is, "does not exist" in the network, according to the feature variables associated with this link. The feature variables associated with a link L_{s_i,s_j} are found by taking the two supplier nodes that the link connects to and extracting $w(O_{s_i,s_j})$, $w(B_{s_i,s_j})$, $w(C_{s_i,s_j})$, D_{S_i} as described previously.

The feature variables and dependent variable form the L dataset, which is divided into a training set L_{train} and a test set L_{test}. To train the classifier, 70% of the links are put in L_{train} and the remaining 30% in L_{test}. We then try to predict links in this 30% by using only the feature variables associated with the nodes. Each predicted link that is found within the test set is considered as a correct prediction, and each predicted link not found in L_{test} is considered a mistake.

It is important to note that link prediction problems, when characterised as a binary classification problem, result in class imbalance. This means that the negative class, that is, links that do not exist in the network, is much larger than the positive class, that is, links that do exist. In a supply network of 200 nodes and 500 relations, the positive class would be 500; whereas the negative class would be $N(N-1)/2-500 = 19{,}400$. Class imbalance is a complicating factor, because it results in a tendency towards false positive classification error, as there are many more instances of the negative class. The implication is that predicting the small class (existence of a link) is more difficult than the large class (nonexisting links) because the biggest source of training data for the algorithm is on the large class. However, it is the small class that is the main target of the predictive process.

Similar to previous researchers, to mitigate this issue, we bias L_{train} by including a random selection of equal numbers of the positive and negative class [19–22]. This is also called oversampling, because we add more entities from the small class to balance out the training process. This helps the classifier learn how to differentiate effects of the feature variables on the dependent variable more accurately. Thus, we generate L^{*}_{train} by oversampling from L_{train}.

In the past, machine learning for link prediction had limited success when data on each instance was limited to topological features [4]. However, in our case, we combine domain specific relational patterns in addition to one topological feature, that of degree. In addition, using relational data means that we develop the understanding of what features are useful in connecting suppliers together, rather than black-box approaches that extract features automatically.

4. Experimental Results

4.1. Automotive Dataset. To illustrate SNLP, we use data from a private automotive industry database (Marklines Automotive Information Platform). The database collects data populated through surveys sent to automotive supplier firms and is primarily used by buyers to search for suppliers

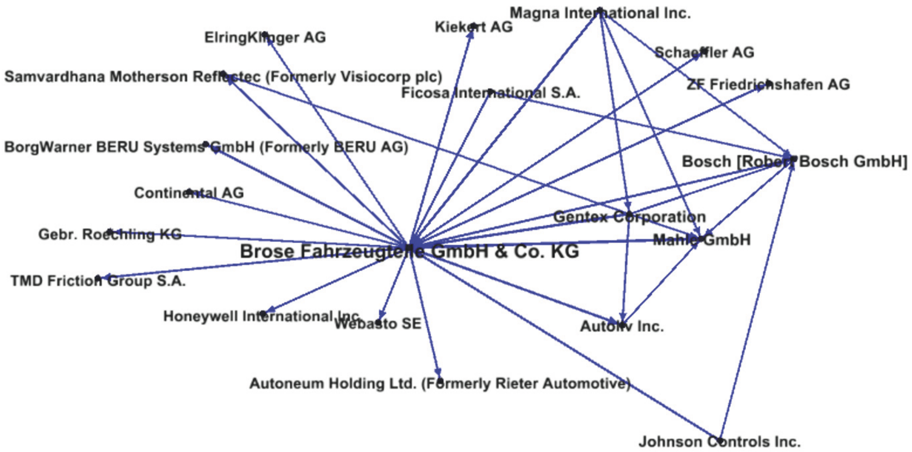

Magna International Inc.
Kiekert AG
ElringKlinger AG
Schaeffler AG
Samvardhana Motherson Reflectec (Formerly Visiocorp plc)
ZF Friedrichshafen AG
Ficosa International S.A.
BorgWarner BERU Systems GmbH (Formerly BERU AG)
Bosch [Robert Bosch GmbH]
Continental AG
Gentex Corporation
Gebr. Roechling KG
Mahle GmbH
Brose Fahrzeugteile GmbH & Co. KG
TMD Friction Group S.A.
Honeywell International Inc.
Webasto SE
Autoliv Inc.
Autoneum Holding Ltd. (Formerly Rieter Automotive)
Johnson Controls Inc.

Figure 2: Top 20 buyer associations in the JLR Network.

Table 1: Number of training instances in three empirical supply networks.

	JLR	Saab	Volvo
Number of nodes	1070	1021	1828
Link	2491	2558	5272
No link	1064871	1038790	3334240
Link : no-link ratio	1 : 427	1 : 406	1 : 632
Average links per node	2.33	2.51	2.88

and suppliers to advertise their capabilities. The data are agglomerative in that once a supplier has identified itself as a supplier to a certain firm, it will remain so, unless either the customer firm or the supplier firm requests a removal of the relationship from the database. Therefore, the data are cross-sectional and might show relationships that are not continuous, although most data were gathered after 2007.

Three supply networks were created by querying focal manufacturer firms Jaguar Land Rover (JLR), Saab, and Volvo. This resulted in the identification of first tier suppliers of the manufacturer. The suppliers were then iteratively queried which resulted in the suppliers of suppliers. The iterative querying continued until no further supply layers were found in the network.

The number of nodes, links, and nonlinks is given in Table 1.

4.2. Link Prediction. Some example associations extracted from the Jaguar Land Rover automotive network include Figure 2, which shows the top 20 buyer associated suppliers. For example, buyers that buy from Magna International are also likely to buy from Mahle GmbH. Therefore, a manufacturer which has a supplier in its network buying from Magna International might predict that the supplier is also likely to buy from Mahle GmbH.

Figure 3 shows the top 20 associated products. For example, pipes have a high dependency relationship with seals, and door trims have a high dependency relationship with fabric/leather. The relationships make sense and point to a rudimentary structural relationship gathered through supply relationships in the original network.

Figure 4 shows the top 20 suppliers who share the highest number of product associations. For example, Autotube Manufacturing Ltd. sells products that are most likely to have a dependency to products produced by a number of companies including Robert Bosch and Continental AG.

Figure 5 shows the top 20 suppliers that have competition associations between them; these are the suppliers who have the highest number of overlapping products in their portfolio. It is interesting that a few of the nodes that are in the top 20 also are in the buyers association network. This means that buyers are buying from competing suppliers together, possibly engaging in a multisourcing relationship.

We experimented with two classifiers, namely, Naïve Bayes (NB) and Logistic Regression (LR). The former assumes that the effects of feature variables on a given class are independent of each other. Despite this often inaccurate assumption, the NB classifier is useful in practice. In particular, the decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one-dimensional distribution. This helps alleviate problems stemming from the curse of dimensionality, such as the need for data sets that scale exponentially with the number of features. While NB often fails to produce a good estimate for accurate class probabilities, this may not be a requirement for many applications where only the classification is required and not the probability associated with it. This is true regardless of whether the probability estimate is inaccurate.

LR measures the relationship between a categorical dependent variable and independent (feature) variables by estimating probabilities using a logistic function. The dependent variable is modeled as a linear combination of the feature variables. Although in NB the weights for each variable are determined independently; whereas in LR weights are set together. In addition to being popular classifiers, we opted for the use of both NB and LR in order to test the independence assumption.

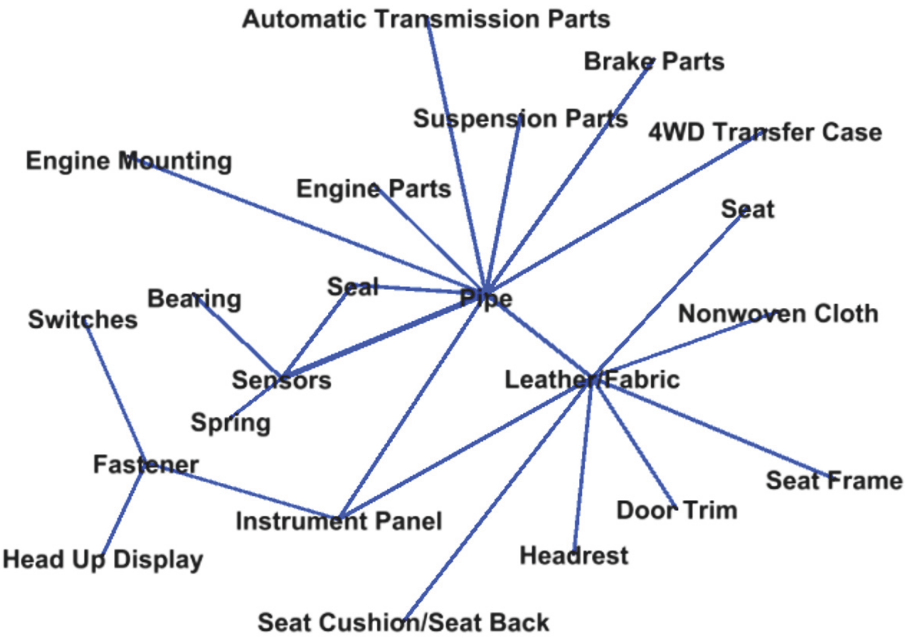


Figure 3: Top 20 product outsourcing associations in the JLR Network, product view.

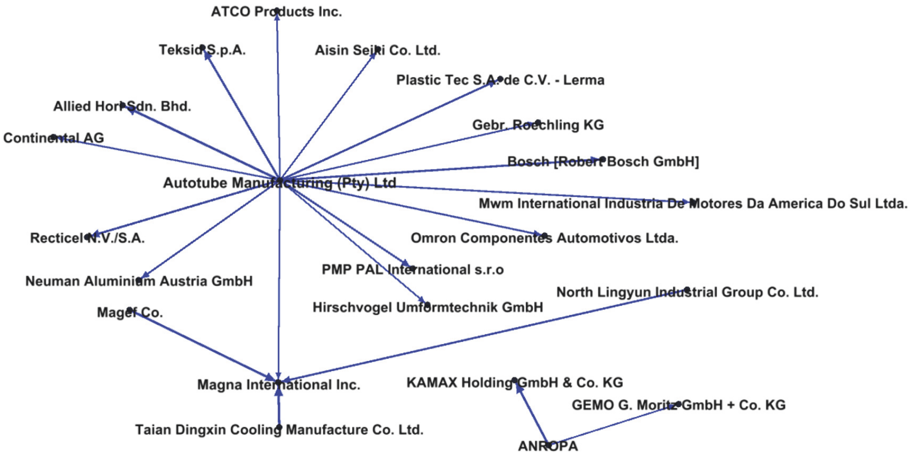

ATCO Products Inc.
Teksid S.p.A.
Aisin Seiki Co. Ltd.
Plastic Tec S.A. de C.V. - Lerma
Allied Hori Sdn. Bhd.
Gebr. Roechling KG
Continental AG
Bosch [Robert Bosch GmbH]
Autotube Manufacturing (Pty) Ltd
Mwm International Industria De Motores Da America Do Sul Ltda.
Recticel N.V./S.A.
Omron Componentes Automotivos Ltda.
PMP PAL International s.r.o
Neuman Aluminium Austria GmbH
North Lingyun Industrial Group Co. Ltd.
Magef Co.
Hirschvogel Umformtechnik GmbH
Magna International Inc.
KAMAX Holding GmbH & Co. KG
GEMO G. Moritz GmbH + Co. KG
Taian Dingxin Cooling Manufacture Co. Ltd.
ANROPA

Figure 4: Top 20 suppliers with highest product outsourcing association in the JLR Network.

4.3. Performance Evaluation. Once the classifier has been trained with the dataset; performance metrics need to be applied to determine the performance of the classifier. The most frequently used metrics to evaluate the performance of classifiers are based on accuracy. In order to calculate accuracy we build a confusion matrix, which displays the correctly and incorrectly predicted instances with predicted instances in the column and actual instances in the row. The class recall is the percentage of correct predictions within each class and the accuracy is the average of correct predictions over the two classes

In addition, for data sets with class imbalance, the most frequently used evaluation metric is the Receiver Operating Characteristic (ROC) curve and the area under the curve (AUC) measure derived from ROC [23]. The ROC curve plots the true positive rate (the fraction of true positives out of the positives) versus the false positive rate (the fraction of false positives out of the negatives) for a binary classifier as its confidence threshold is varied. ROC curves are calculated by first ordering the classified examples by confidence. Afterwards, all the examples are ordered with decreasing confidence to plot the false positive rate on the x-axis and the true positive rate on the y-axis. The result is a line which is a straight diagonal if the model is merely guessing and the more and more a curve moves towards the top left corner, the better the model gets. The AUC measures the area below the curve in order to compare the overall predictive performance of two different curves. A value larger than 0.5 points to the fact that

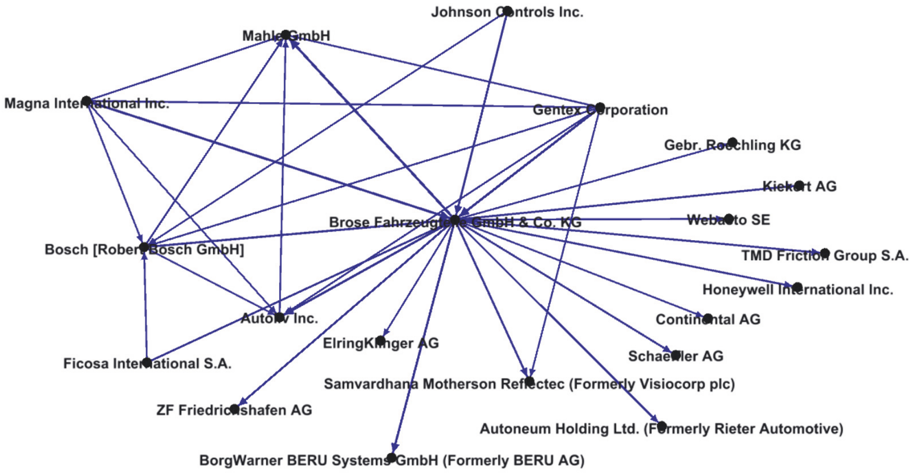
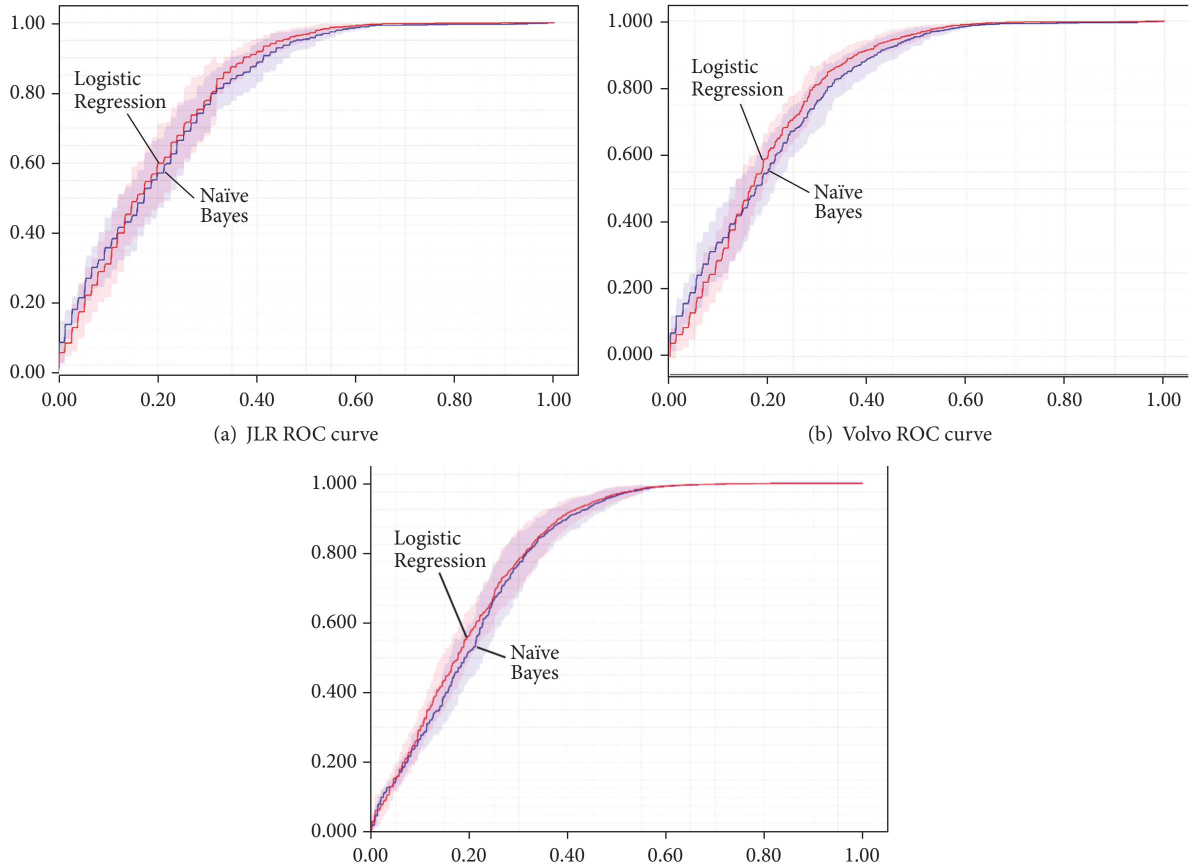

Johnson Controls Inc.
Mahle GmbH
Magna International Inc.
Gentex Corporation
Gebr. Roechling KG
Kiekert AG
Webasto SE
Brose Fahrzeugteile GmbH & Co. KG
Bosch [Robert Bosch GmbH]
TMD Friction Group S.A.
Honeywell International Inc.
Continental AG
Autoliv Inc.
ElringKlinger AG
Schaeffler AG
Ficosa International S.A.
Samvardhana Motherson Reflectec (Formerly Visiocorp plc)
ZF Friedrichshafen AG
Autoneum Holding Ltd. (Formerly Rieter Automotive)
BorgWarner BERU Systems GmbH (Formerly BERU AG)

Figure 5: Top 20 suppliers with highest competitor association in the JLR Network.



(a) JLR ROC curve

(b) Volvo ROC curve

(c) Saab ROC curve

Figure 6: ROC curves for classifier results in three supply networks.

the algorithm is predicting significantly better than a random guess. Table 2 shows the confusion matrices and AUC for all three networks, and the two classifiers, while Figure 6 shows the ROC curves. Both classifiers across all three test cases show similar accuracy and AUC levels. The AUC in all cases indicate significantly better than random guess. True negatives are in general better predicted than true positives; however, the overall performance is encouraging. The lowest ratios of true positive occur in the Volvo and Saab Naïve Bayes case, which show a clear imbalance towards true negatives.

Table 2: Comparison of predictive accuracy.

(a) Confusion Matrices for each case study, AUC and Accuracy

Algorithm	Logistic Regression		Naïve Bayes	
	Link	No-link	Link	No link
		JLR		
Predicted true	501	45878	573	117164
Predicted false	248	273582	176	202296
Class recall	66.89%	85.64%	76.50%	63.32%
AUC	0.81		0.80	
Accuracy	0.76		0.70	
		Saab		
Predicted true	590	114585	405	18877
Predicted false	159	197070	344	292778
Class recall	78.77%	63.23%	54.07%	93.94%
AUC	0.81		0.80	
Accuracy	0.71		0.74	
		Volvo		
Predicted true	1040	134402	834	42997
Predicted false	547	865865	753	957270
Class recall	65.53%	86.56%	52.55%	95.70%
AUC	0.81		0.79	
Accuracy	0.76		0.74	

(b) PR-AUC and PR-CAUC

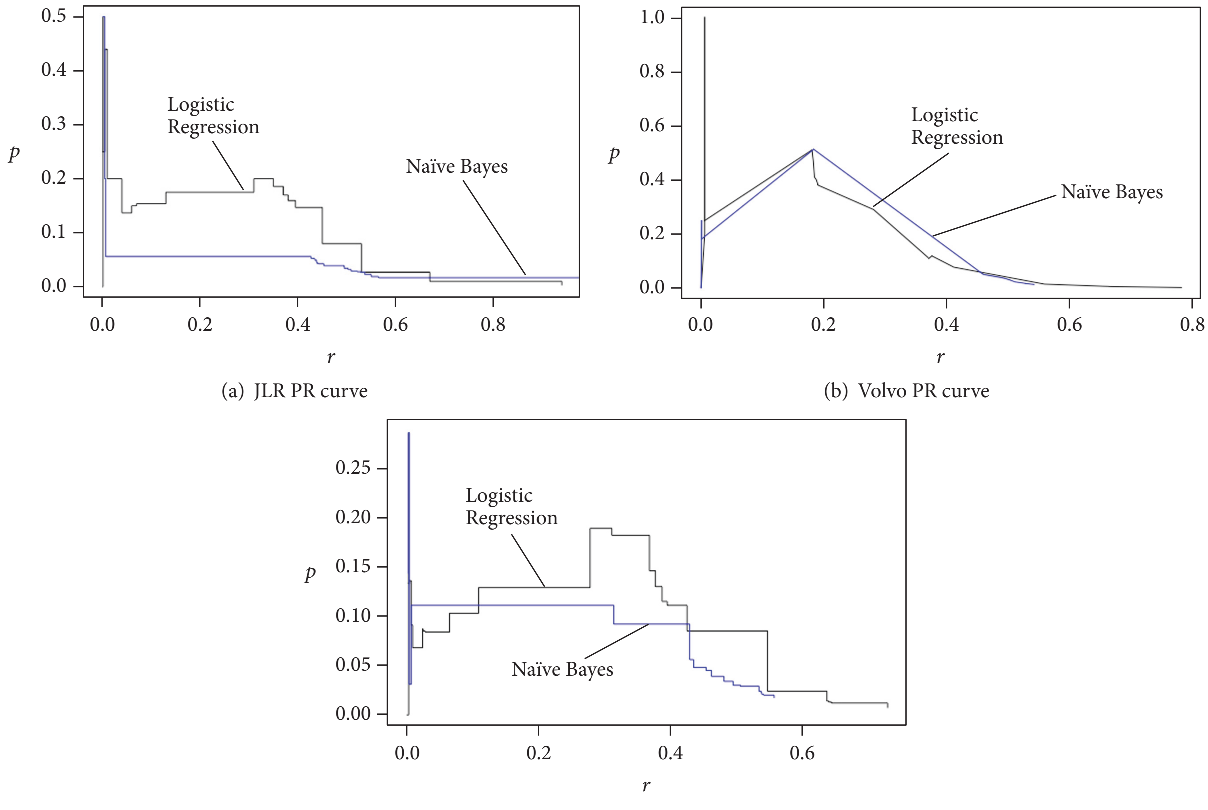
	Volvo	JLR	Saab
Logistic Regression			
PR-AUC	0.140	0.085	0.073
PR-CAUC	0.113	0.024	0.012
Naïve Bayes			
PR-AUC	0.143	0.033	0.044
PR-CAUC	0.063	0.002	0.0004

(c) PR-AUC and PR-CAUC with increased training sizes

	Volvo	JLR	Saab
$n = 250$			
PR-AUC	0.1	0.073	0.073
PR-CAUC	0.9	0.019	0.009
$n = 500$			
PR-AUC	0.14	0.085	0.073
PR-CAUC	0.113	0.024	0.012
$n = 1000$			
PR-AUC	0.16	0.11	0.11
PR-CAUC	0.12	0.04	0.03

Overall, LR offers better prediction than NB according to the ROC and AUC measures, possibly due to its underlying assumption of feature independence.

One should note that although previous works in link prediction used AUC and accuracy in performance assessment (e.g., [8, 10, 17, 24], Fire et al. 2013), there is some recent debate on the usefulness of this metric [4, 25]. It has been recognised that link prediction problems yield extremely low precisions due to class imbalance, because the smallest rate of acceptance will amount to a large number of wrongfully predicted edges leading to a needle in a haystack type problem. It is argued that in large networks the absence of a particular link is an irrelevant issue, particularly when there are millions of possible edges; we are interested in the existence of links instead. In this respect, Precision Recall Curves (PR) and the area under the PR curve (PR-AUC) are deemed to be more meaningful performance measures than AUC because PR ignores true negatives, which are not



(a) JLR PR curve

(b) Volvo PR curve

(c) Saab PR curve

Figure 7: PR curves for classifier results in three supply networks.

relevant to the problem. The PR curve plots the precision on x-axis and recall on the y-axis and hence does not show correct classifications for the negative class. Different to ROC, the PR curve would display random classifier performance as a straight line on the x-axis, as precision in imbalanced data would be close to zero (Yang et al. 2014). Garcia-Gasulla [4] further proposes the PR-CAUC (constrained AUC) measure, which focuses on applicability. The PR-CAUC measure is obtained by calculating the AUC of the subcurve, where the number of nonexisting links mistakenly accepted by the classifier is equal to or lower than the total number of links in the original network. The measure therefore calculates a portion of the PR-AUC starting from the left of the curve, stopping when too many mistakes are done.

The reasoning behind PR-CAUC is that link prediction in real life cases does need to classify most edges correctly in order to be successful but needs only to correctly inform about the existence of a significant set of links with high certainty to become useful. Hence, instead of building two classes of links (i.e., existing and not existing), the prediction should aim at approximating links whose "existence is well founded and, most importantly, understood [4]." We agree with this point of view, because in a supply network it would be unlikely that there are a significant number, that is, in the order of hundreds, of links invisible to the manufacturer. Hence, the manufacturer would want to predict the existence of a small but significant set of dependencies with high confidence rather than all possible links.

Because the sampling process has inherent randomness in the training process, the training and testing procedure was repeated 30 times with each algorithm and average performance measures were taken. In each case, training and test links were randomly selected. Figure 7 displays the PR curves next to each ROC curve associated with the three test cases, whereas Table 2(b) displays the PR-AUC and PR-CAUC for each test case. The Volvo test case yields the most promising results; the Logistic Regression classifier provides about 300 correct links (recall = 0.19) with a 50% precision. The JLR and Saab test cases yield lower precision, possibly due to the lower number of links that could be used to train the classifiers. Similarly, the PR-AUC and PR-CAUC values in these two test cases are lower than Volvo. Comparison of PR-CAUC value with those of Garcia-Gasulla ([4], 2016) test cases on 9 different imbalanced graphs shows that our results are competitive.

It is interesting that according to the PR-AUC measure Naïve Bayes (NB) performs better than Logistic Regression (LR) by a small margin, but according to the PR-CAUC measure NB performs better than LR by a large margin. This difference is evident in both the ROC and the PR curves, where the recall threshold of CAUC (i.e., the CAUC stops considering the PR curve), the LR is higher than the NB

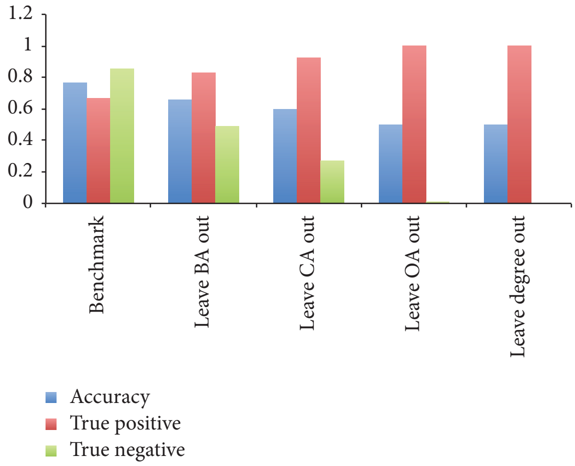


Figure 8: Reducing accuracy as features are removed from the training set (JLR case).

curve. This means that LR performs significantly better for high confidence predictions (the first few thousands of links), while NB performs better at trying to recover every single missing link. These results are consistent with the other two cases, where LR outperforms NB.

The effect of training dataset size is shown in Table 2(c), for the LR case across the three automotive networks, where the samples for true and false cases were increased from 250 to 1000. While the effect is expected as the size of the dataset results in increased PR-AUC and PR-CAUC, the increase is only slight. In the absence of more data, we cannot test whether a more significant increase in training dataset size would result in much higher performance; however, this result hints that at least slightly better performance is expected.

In order to test the impact of different feature variables on the accuracy of the model, we followed a procedure where each feature variable was left out and compared to the benchmark result in which all variables were included in the predictive process. Figure 8 displays the results of the JLR case. Other test cases gave similar results.

Overall accuracy decreases most by leaving out D_{S_i}, $w(O_{s_i,s_j})$, $w(C_{s_i,s_j})$, and $w(B_{s_i,s_j})$, respectively; meaning that the predictive accuracy is maximised by degree and outsourcing associations. These two features hold the most information on what constitutes a true negative; because when they are left out, the classifier struggles to classify links that do not exist. So although the classifier still manages to detect links that exist, overall accuracy is diminished. It is interesting that the topological feature variable is as powerful as the relational variables.

5. Conclusions

The SNLP approach presented here offers manufacturers an opportunity to reduce the risks associated with the lack of visibility of their supply network using only the minimal amount of data they have available. Results from the experimental testing of SNLP suggest first that the extracted features can be successful in prediction and that predictive accuracy is maximised by employing data on dependency relations between products and the out-degree of suppliers. Thus, we contribute to extant literature by first providing an alternative, complimentary approach to the detection of procurement interdependencies in supply networks that do not rely on suppliers sharing data.

It should, however, be noted that the method produced in this study has been applied only to the automotive sector. While we have shown that the combination of graph mining and machine learning can be powerful, the specific features that inform a prediction might not prove applicable to other industries. For example, we found that several suppliers that have overlapping product portfolios also are bought together by manufacturers, pointing to a multisourcing relationship. Thus, the competition association might not prove informative in industries where competition is low, such as aerospace. Furthermore, the outsourcing association might only prove informative in industries where the end product contains sufficiently large number of parts that are distributed across the network. The extent to which this feature would be useful can also be gauged by the level of vertical integration an industry typically displays. Practitioners therefore need to think about features relevant to the industrial sector under query before applying SNLP.

The study has further limitations, which in turn provide avenues for future research. First of these are limitations on data. The dataset we used might include missing links or links that do not exist anymore. The second limitation is related to methods. The classification algorithms used were limited to Logistic Regression and Naïve Bayes. LR performed significantly better than NB for high confidence predictions, while NB performed better at trying to recover every single missing link. We would like to explore these issues and findings further by testing the approach with other datasets and prediction algorithms. Similar to other link prediction problems in large-scale graphs, our approach suffers from class imbalance; the nonexisting link class is much higher in number than the existing class, limiting precision levels. In this respect we found the PR-CAUC measure based on applicability a useful metric to evaluate outcomes. Using the PR-CAUC measure as an objective function for the training algorithm could be useful in improving algorithm performance.

An interesting stream of exploration to strengthen precision could include path-based dependencies. The extracted outsourcing relations are currently binary in that they point to dependencies between two products such as coffee granules and coffee jars or coffee beans and coffee granules. A better approach could be the extraction of a "pathway" of dependency, such as coffee beans to coffee granules to coffee jars, which could inform product structures, potentially leading to more informed predictions. The possibility of secondary or indirect prediction is also intriguing. For example, the manufacturer might not have full visibility of suppliers' product portfolio or capabilities. The detection of missing products could lead to better prediction of missing links. Finally, additional features such as supplier location could be explored.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] Z. E. Tang, M. Goetschalckx, and L. McGinnis, “Modeling-based design of strategic supply chain networks for aircraft manufacturing,” in *Proceedings of the 11th Annual Conference on Systems Engineering Research, CSER 2013*, pp. 611–620, USA, March 2013.

[2] D. Lin, “An information-theoretic definition of similarity,” in *Proceedings of the 15th Int. Conf. on Machine Learning*, Madison, Wisconsin, USA, 1998.

[3] S. B. Kotsiantis, “Supervised machine learning: A review of classification techniques,” in *Emerging Artificial Intelligence Applications in Computer Engineering*, I. Maglogiannis et al., Ed., pp. 3–24, 2007.

[4] D. Garcia-Gasulla, *Link prediction in large-scale directed graphs, PhD thesis Universitat Politecnica de Catalunya*, Barcelona, 2015.

[5] D. Liben-Nowell and J. Kleinberg, “The link prediction problem for social networks,” in *Proceedings of the 12th ACM International Conference on Information and Knowledge Management (CIKM ’03)*, pp. 556–559, ACM, November 2003.

[6] L. Katz, “A new status index derived from sociometric analysis,” *Psychometrika*, vol. 18, no. 1, pp. 39–43, 1953.

[7] E. A. Leicht, P. Holme, and M. E. J. Newman, “Vertex similarity in networks,” *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 73, no. 2, Article ID 026120, 2006.

[8] L. Lü, C.-H. Jin, and T. Zhou, “Similarity index based on local paths for link prediction of complex networks,” *Physical Review E: Statistical, Nonlinear, and Soft Matter Physics*, vol. 80, no. 4, Article ID 046122, 2009.

[9] Z. Liu, Q.-M. Zhang, L. Lü, and T. Zhou, “Link prediction in complex networks: a local naïve Bayes model,” *EPL (Europhysics Letters)*, vol. 96, no. 4, Article ID 48007, 2011.

[10] A. Clauset, C. Moore, and M. E. J. Newman, “Hierarchical structure and the prediction of missing links in networks,” *Nature*, vol. 453, no. 7191, pp. 98–101, 2008.

[11] H. P. Thadakamalla, U. N. Raghavan, S. Kumara, and R. Albert, “Survivability of multiagent-based supply networks: a topological perspective,” *IEEE Intelligent Systems*, vol. 19, no. 5, pp. 24–31, 2004.

[12] E. J. S. Hearnshaw and M. M. J. Wilson, “A complex network approach to supply chain network theory,” *International Journal of Operations & Production Management*, vol. 33, no. 4, pp. 442–469, 2013.

[13] A. Brintrup, A. Ledwoch, and J. Barros, “Topological robustness of the global automotive industry,” *Logistics Research*, vol. 9, no. 1, article no. 1, pp. 1–17, 2016.

[14] A. Brintrup, Y. Wang, and A. Tiwari, “Supply Networks as Complex Systems: A Network-Science-Based Characterization,” *IEEE Systems Journal*, vol. 11, no. 4, pp. 2170–2181, 2017.

[15] N. Friedman, L. Getoor, D. Koller, and A. Pfeffer, “Learning Probabilistic relational models,” in *Proceedings of the 16th Int. Joint Conference on Artificial Intelligence*, Stockholm, Sweden, 1999.

[16] K. Yu, W. Chu, S. Yu, V. Tresp, and Z. Xu, “Stochastic Relational Models for Discriminative Link Prediction,” in *Proceedings of Neural Information Processing Systems*, p. 1553, Cambridge MA, 2006.

[17] C. C. Aggarwal, Y. Xie, and P. S. Yu, “A framework for dynamic link prediction in heterogeneous networks,” *Statistical Analysis and Data Mining*, vol. 7, no. 1, pp. 14–33, 2014.

[18] M. Al-Hassan, V. Chaoji, S. Salem, and M. J. Zaki, “Link prediction using supervised learning,” *In SDM Workshop on Al-Link Analysis, Counter-terrorism and Security*, 2006.

[19] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[20] G. M. Weiss, “Mining with rarity: a unifying framework,” *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004.

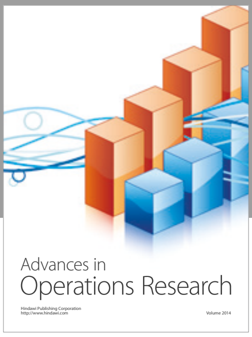
[21] X.-Y. Liu, J. Wu, and Z.-H. Zhou, “Exploratory undersampling for class-imbalance learning,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 2, pp. 539–550, 2009.

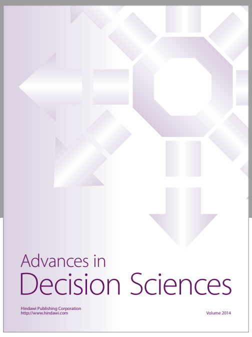
[22] M. Wasikowski and X.-W. Chen, “Combating the small sample class imbalance problem using feature selection,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1388–1400, 2010.

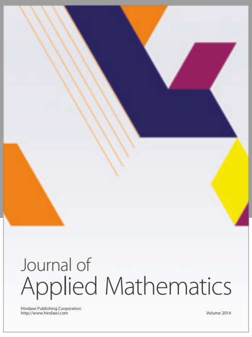
[23] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

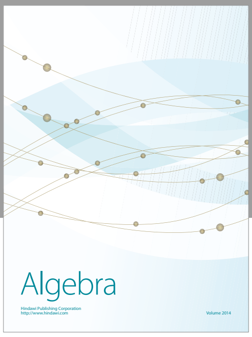
[24] M. Nickel, V. Tresp, and H. Kriegel, “A Three-Way Model for Collective Learning on Multi-Relational Data,” in *Proc. of the 28th Int. Conf. on Machine Learning*, Bellevue,WA, USA, 2011.

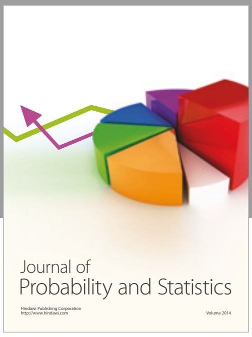
[25] D. Garcia-Gasulla, U. Cortes, E. Ayguade, and J. Labarta, “Evaluating Link Prediction on Large Graphs,” in *Proceedings of 18th Int. Conf. of the Catalan Association for Artificial Intelligence*, pp. 90–99, 2015.


Advances in
Operations Research

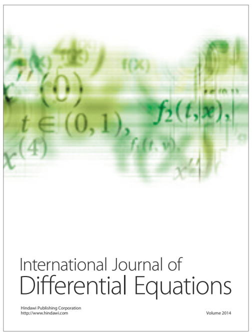

Advances in
Decision Sciences

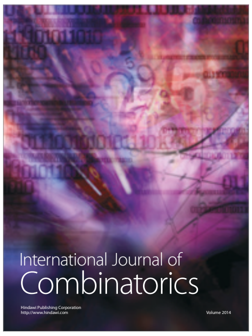

Journal of
Applied Mathematics

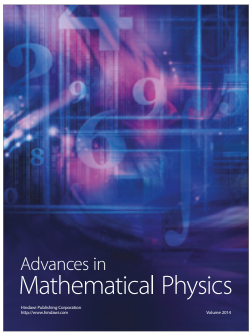

Algebra

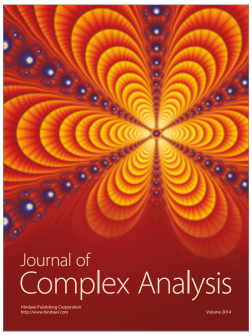

Journal of
Probability and Statistics

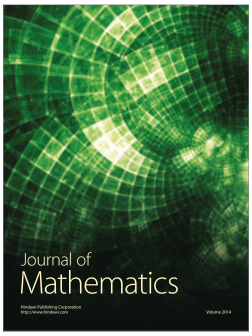
The Scientific
World Journal

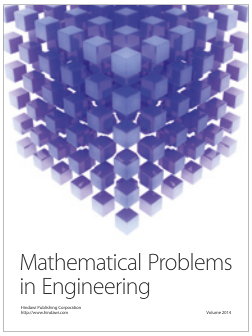

International Journal of
Differential Equations

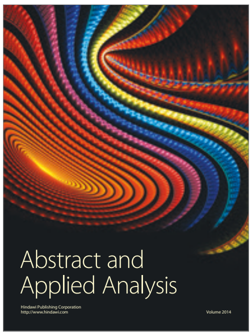

International Journal of
Combinatorics

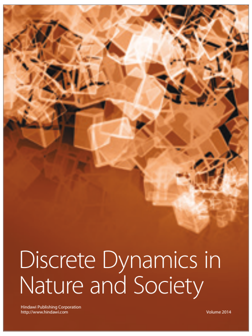

Advances in
Mathematical Physics

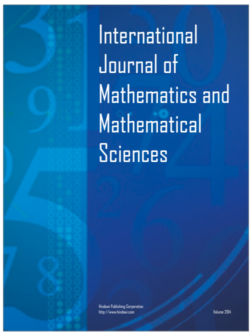

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

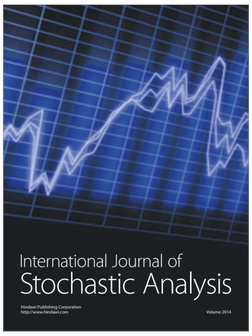

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization