Internal Link Prediction in Early Stage using External Network

Honghao Wei
Stanford University
weihh16
weihh16@stanford.edu

Yiwei Zhao
Stanford University
ywzhao
ywzhao@stanford.edu

Junjie Ke
Stanford University
junjiek
junjiek@stanford.edu

Abstract

Modern E-commerce recommendation systems use co-purchase history to increase user conversion and engagement. However, without sufficient purchase history, analysis with co-purchase recommendation is difficult, and involves cold-start issue in practice. In this project, we are motivated to explore ways to utilize the IMDB casting network to enhance the Amazon co-purchase network inference in cold-start and early post cold-start stage with insufficient purchase history. To tackle cold-start issue, we perform a multi-start supervised random walk over external network, derived from IMDB casting network. We then augment the external network by adding internal network edges and rank new candidates based on proximity features ensemble with cold-start results. Our methods outperform the conventional benchmark models in both cold-start and early post cold-start stages and prove external network to be extremely useful in early stage link inference.

*(**Note we are not challenging the state-of-the-art link prediction methodologies but to investigate brand-new ways of tackling cold-start and early stage internal link prediction using external network.**)*

1. Introduction

Modern E-commerce and online shopping become increasingly popular in recent decade. One success factor is to build a smart recommendation system based on historical co-purchase network data. By exploitation to uncovered correlation of purchases, e-commerce recommendation system creates a better understanding of user habits, enhance buying experience and increase sales for more profits. Despite the huge success when user and purchase history accumulates, there are difficulties in such way of recommendation, such as **Cold-start** issue. In practice, it is less likely to have an objective recommendation when new products on sale lack purchase history. In addition, the buying needs of existing commodities, especially Non-consumables, tends to decrease while latest sale products is more likely to arouse user interests and curiosity with conversion to purchase and engagement. Therefore, it is necessary to solve cold-start issue and early stage prediction in co-purchase inference.

In this project, we investigate the prediction of co-purchase in videos and DVD items on Amazon in particular without purchase history or with only limited records in early stage. We notice the external review and casting information is extremely helpful in co-purchase link inference. Shown in Fig.1, given a user is query for results of video 'Edward Scissorhands', Amazon suggests co-purchase relationship with 'Sweeney Todd', 'Alice in Wonderland' and 'Dark Shadows'. All these three movies share the same starring (Johnny Depp), same director (Tim Burton) and same genres (Fantasy) with the video user interested in. We find similar casting information in IMDB for videos display on Amazon. To take advantage of external information, we could alleviate cold-start situation, in which item-item purchase correlation could be inferred via casting network instead of purchase history in early stage with limited buying records, and encourage purchase of newly release items to enhance buying experiences and result in better profits.

To better formulate the co-purchase prediction problem, we define two types of network, internal and external.

- **Internal network** refers to the target network, such as Amazon purchase history network in which co-purchase relationship is predicted.
- **External network** contains the same nodes in internal network but the edges stands for different interaction. For example, in IMDB network, the link represents same actors, actresses or directors.

We state our work is not to challenge the most advanced general link prediction methodologies. We address our efforts to solve cold-start and early stage link prediction using external network with simplicity in acquisition. With accumulation of purchase history, Our proposed methods is very likely to underperform the state-of-the-art models in co-purchase inference. However, they are proved to be extremely useful in early stages including cold-start phase,

and outperform greatly than the widely-used benchmarks by raising 22% precision.

2. Related Works

In previous works of link prediction, most of works uses information from target network and perform the inference. Some works use external information instead but limitation exists. Apart from conventional ways of link prediction, some recent works tend to adapt machine learning methods using features extracted from targeted network. In these works, the nodes in targeted network is divided into training set and test set, and features such as edge and neighbors in training set are used to tune the parameter in link prediction problem. We look into methods and limitations in these three types of link inference problems and give our own insight in co-purchase link inference.

Link prediction with internal network information

The most conventional and robust way of link prediction is to extract topological structure from internal network and use node proximity to perform direct measure[8, 10]. However, these methods depend on information in internal network and has no means to avoid cold-start problem. In our baseline, we would demonstrate the disadvantage in co-purchase inference.

Link prediction with external network information

Besides link prediction with internal network information, some other methods take advantage of external network information[11, 15]. External information such as facebook network and profiles [15] are utilized for aid of link inference. However, few of works propose to merge the internal and external network together to further exploit the correlation of purchase and product features. In our work, we would adopt several ways of network merging strategy to utilize external network features in link prediction of internal network.

Link prediction with supervised machine learning methods

In previous works, link prediction is studied as a supervised learning task[1, 5, 4, 9]. They identified a set of features key to performance in supervised learning setup, which are easy to compute but surprisingly effective in solving link inference problem. In our work, we would explain the effectiveness of key features of network consisting of both internal and external information, and compare different classes of supervised learning algorithms in terms of their prediction performance using various performance metrics, such as accuracy, precision-recall, F-values, squared error, and etc.

Link prediction with cold-start

Different methods have been used to predict the links when the node is newly added, with its network information is totally missing while some other information regarding the nodes is available. Compared with the regular link prediction problem, prediction with cold start is extremely hard since when a new node arrives, no network information is available and we need to utilize external network information to make prediction, which required deeper understanding of the network itself and more insight methods. For the cold-start problem, people usually used probabilistic model to to learn the structure of the known system. Schein et al. [13] used a model that combines both collaborative and content information in model fitting. This folding in algorithm allows them to make predictions for unrated items by using content data and successfully solved the cold-start issue. For the link prediction, Leroy et al. [6] proposed a two-phase method, which is based on the bootstrap probabilistic graph. For the first phase of their method, according to the network they already have, it generates an implicit social network under the form of a probabilistic graph. Using this generated probabilistic graph, for the second phase, it applies probabilistic graph-based measures to produce the final prediction for the newly arrived nodes. Zhang et al. used the social media information as an external data to predict the links, which also helps to avoid the cold-start issue [15]. Wang et al. [14] developed an effective approach which can establish connections between non-topological and topological information. In the approach, topological information is first extracted using a latent-feature representation model, then a logistic model is proposed to establish the connections between topological and non-topological information and finally predict links between cold-start users and existing users is calculated.

3. Datasets and Network Merging

In this section, we display the demographic results of both internal network (Amazon co-purchase network) and external network (IMDB casting network).

3.1. Internal Network

In order to evaluate how well we could predict co-purchase link and compare results with the state-of-the-art methodologies, we need a ground truth data source and other information besides co-purchase. Therefore, we take Amazon Product Review Data with both co-purchase and category information as internal network and specifically restrict ourselves within 'Video and DVD category. We notice there are three key factors to filter inappropriate node in our prediction network [7].

- We would only use videos and DVDs available in IMDB datasets.
- We filter these nodes and make sure top 5 most similar items in 'Videos and DVDs' category since it is likely that the co-purchase link contains products in some categories but not 'Videos and DVDs'.

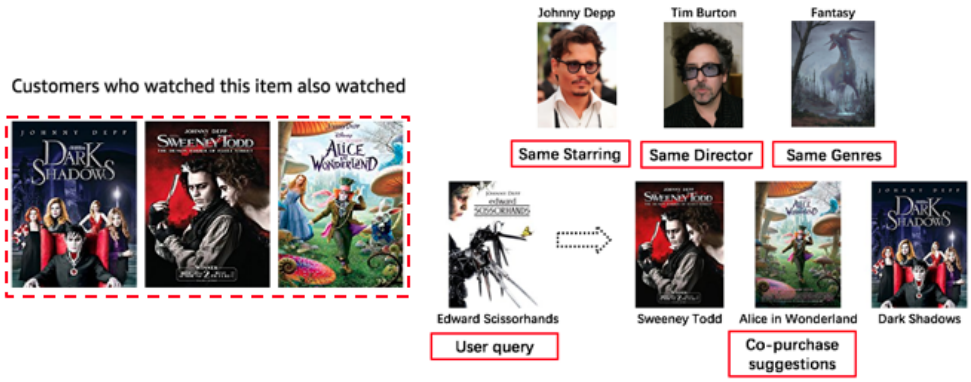

Figure 1: Example of external review information, such as starring, director and genres, impacts user co-purchase behavior.

- We take advantage of product title information to distinguish different movies and TV shows. We find there are different editions, years and ids for the same movie. In this way, we construct the graph to assign movies with same title but different ids, years and edition to the same node and reassign the id for each node.

To sum up, we filter and get 16224 DVDs and Video movies whose title also exist in the IMDB dataset, and label the movies from 1 to 16224 to share index for all the datasets. With same ID, it is easy to map between internal network and external network. There are 31115 co-purchase link in internal network in total.

The demographic features of our internal network is shown below. The clustering coefficient is 0.28109 and the diameter is 20. The degree distribution is showed in Figure 2. There are less than 10 co-purchase neighbors for most of produces on average.

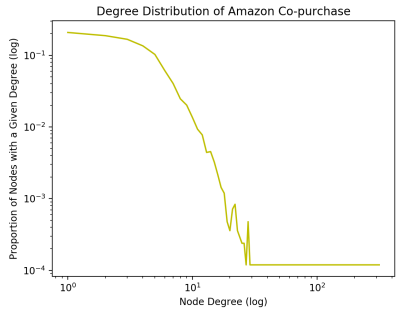

Figure 2: Degree Distribution of Complete Amazon Videos and DVDs Co-purchase Network

3.2. External Network

We use IMDB casting network as complementary information besides internal network features. It contains casting information, including actors, actresses and directors. We filter out IMDB movies that exist in Amazon network and assign the node the same index based on the title. We only add a link between two movies if they share at least one common actor, actress or director. The weight of edge is determined by the total number of common actors, actresses or directors. We filter those castings who star in less than 20 movies to make sure we only take major starring and directors into consideration, who are more likely to impact on audiences' co-purchase behavior. We build four IMDB external networks derived using common actor, actress, director and the union of all three. We experiment all four networks in cold-start stage but only test the last external networks in post cold-start stage due to the complexity in computation and insufficiency in computational resources.

The degree distribution of external network linking movies with common actor, actress and director is showed in Figure 7. We find all three external networks share similar distribution but the average degree in director network is far less than the other two. It is in accordance with the fact that the number of productive directors is less than the number of actors and actresses. It is also interesting to see some movies share link with around 1000 videos due to its connection in leading actors and actresses. We then compute the percentage of co-purchase links covered in the first-degree neighborhood of IMDB external network. As showed in Table 1, we find the percentage of co-purchase covered in IMDB direct neighbors is around 50% and it is reasonable to use IMDB neighbors as candidates for predicting co-purchase links. However, even if we use the union of common actor, actress and director network, it covers only 58.91% of the total co-purchase link in internal network. Therefore, the best theoretical accuracy of our cold-start prediction should be less than 60%. Another bottleneck is that we are selecting 20k true labels from 2 million candidates, resulting in a difficult issue to solve.

Common type	Co-purchase link covered in IMDB neighbors	Percentage in total co-purchase Link	Total IMDB neighbors
Actor	13671	49.09%	2273518
Actress	6200	22.26%	656222
Director	2913	10.46%	82005

Table 1: Percentage of Amazon co-purchase link covered in IMDB neighborhood

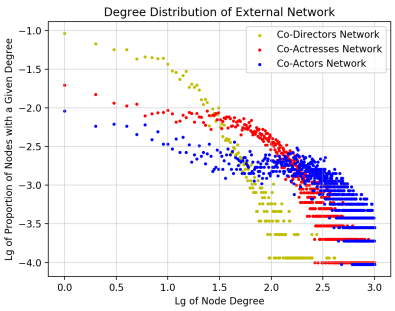

Figure 3: Degree Distribution of External Network Linking Movies with Common Actor, Actress and Director

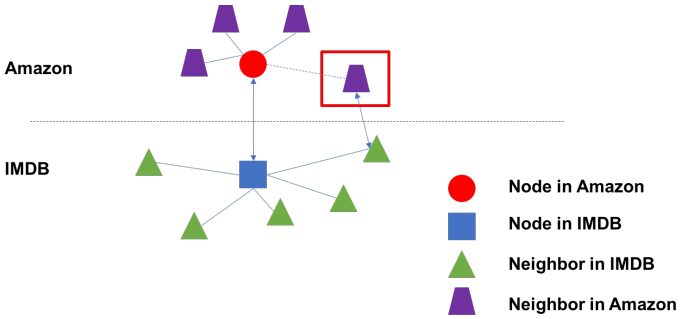

Figure 4: Demonstration about candidates we have in cold-start stage

4. Evaluation and Key features

In this section, we discuss our metrics for evaluation, and key features in network for link prediction.

4.1. Metrics for evaluation

4.1.1 Cold-start evaluation

In cold-start stage, when new products come on sale, there is no purchase history for reference. As showed in Table 1, around 60% of co-purchase link would be covered in the direct neighbors IMDB casting network. Therefore, we use the direct neighbors of common actors, actresses, directors and the union of three to be the candidates of co-purchase candidates in cold-start stage. Therefore, we define the precision and recall as below.

$$P = \frac{TP_0}{TP_0 + FP_0}$$
$$R = \frac{TP_0}{TP_0 + FN_0} \quad (1)$$

, in which TP_0, FP_0 and FN_0 are true positive, false positive and false negative in the direct neighbors of external network. We report precision, recall and f1-measure as evaluation for performance in this stage. As showed in Figure 4, we try to find the co-purchase for a node from its direct neighbors in the IMDB external network, indicated by red rectangle.

4.1.2 Post Cold-start Stage

We add links in the internal network back to external network to represent the accumulation of purchase history in our experiment. We define the first stage with no links in internal network to be cold-start stage, and the rest as post cold-start stage. In post cold-start stage, we use the union of two hop neighbors of internal network, which is also called 'friend of friend', and external network to be the candidate pool. It is reasonable to include links in internal network for reference, since in this stage we know features in internal network as well. Therefore, we define the precision and recall as below.

$$P = \frac{TP_1}{TP_1 + FP_1}$$
$$R = \frac{TP_1}{TP_1 + FN_1} \quad (2)$$

, in which TP_1, FP_1 and FN_1 are true positive, false positive and false negative in the union of two hops neighbors in internal network and direct neighbors in external network. We report precision, recall and f1-measure as evaluation for performance in this stage.

It is important to address that we do not keep the selection within the direct neighbors of external network in post cold-start stage. The first reason is that we have no sequential label of purchase history. More importantly, we do not wish to raise the precision and recall in direct neighbors in external network but result in worse performance of overall candidates in practice. As showed in Figure 5, we try to find

the co-purchase from direct neighbors in external network, indicated by red rectangle, and the two hops neighbors in internal network, indicated by green rectangle.

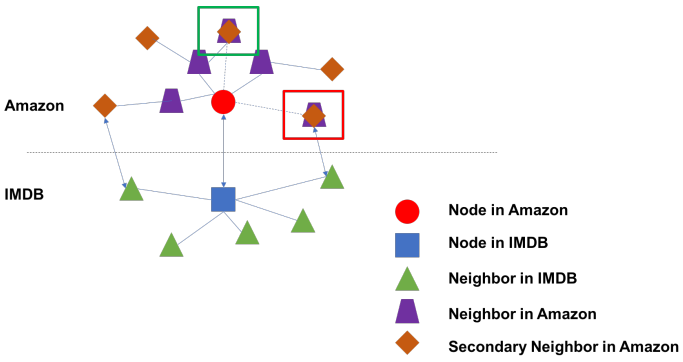

Figure 5: Demonstration about candidates we have in post cold-start stage

4.2. Key features in network for link prediction

We extract following proximity features for prediction in both .

- **Number of Common Neighbors** Measures neighborhood overlap. If both videos are always co-purchased with another videos, then these two are more likely to be a 'also-buy'.
- **Preferential Attachment** The more number of neighbor of two nodes is, the more likely they are to be linked since Better connected nodes are more likely to form more links
- **Shortest Distance** The weighted shortest co-purchase relationship between two nodes
- **Jaccard Coefficient** We compute the fraction of common neighbors to measure the similarity of two nodes.
- **Page Rank Sum** Measures the probability that two videos will be co-purchased on a random walk through the network.
- **Adamic-Adar** We calculate the number of common neighbors, with each neighbor attenuated by log of its degree.

$$\text{Adamic-Adar} = \sum_{z \in \text{common neighbour}} \frac{1}{\log d_z} \qquad (3)$$

5. Methods

In this section, we would introduce the benchmark methods for comparison in both cold-start stage and post cold-start stage and propose our own methods in both stages.

5.1. Benchmark Methods in Cold-start Stage

The following four models are used as baselines for comparison.

Random Model Our raw baseline is random link selection with uniform probability on Amazon co-purchase network.

Category Inference We use non-sequential category information in internal network to infer co-purchase link without. We construct an additional network for Amazon category representation. Noticing Amazon keeps a hierarchical structure for product category in terms of a tree, we build edges in two ways. Firstly, in accordance with strict categorical relationship, we link two nodes only if two products share exact category hierarchy from top to end and all edges share same weight. Another way is to build links if the category of two products partially same. Under this circumstances, weight of edge is determined with similarity of each pair of nodes. For example, all products fall into the category of 'Video and DVD' and thus should share same root of category tree but 'Edward Scissorhands' and 'Sweeney Todd' shares same secondary category 'fantasy', and thus we should assign larger weights to the edge between them. There are 25832 different categories and 1688592 edges created.

For each node, we choose the top3 or top5 nodes which share most similar category hierarchy. If there are tiers, we choose randomly. Given E as the ultimate number of edges selected and N as total number of nodes, we have $\frac{3N}{2} \leq E \leq 3N$.

5.2. Multi-start Supervised Random Walk

To model the random walk transition probability on the external IMDB graph, for each edge (u, v) in G, we compute the strength a_{uv}, which is a function of weight w and edge feature vector ϕ_{uv}, i.e. $a_{uv} = f_w(\phi_{uv})$ [3]. The features of each link is selected to be its Common neighbors, Jaccard coefficient, Adamic/Adar and PageRank. We will use the learning algorithm in the training phase to find the best weight w_{opt}.

To predict a node s's new neighbors, utilizing the edge strengths obtained through features and the current weight w, we run a page rank algorithm to find the stationary distribution p for each node u. Then the nodes are ordered by it page rank scores and top ranked nodes are then predicted as destinations of future links of the s.

After obtaining a predictor for the new links of a node, we can then leverage the labeled links to learn the optimized weight w. So the overall procedure of this method can be described as: Firstly, using the existing links of amazon co-purchase network as label, we used supervised random walk on external IMDB network to learn a best weight w_{opt}. Then for a new node first appear in amazon co-purchase network, we can use the learned weight to predict the top link

candidates from the IMDB network.

In the following sections, we describe the details of our supervised optimization method and how we extend our method to train the global weight.

5.2.1 Optimization Goal

Assume the $p_x(w)$ denotes the Pagerank of point x under the weight w, for each node s, we formalize our optimization goal as in [3]:

$$\min_w F_s(w) = \lambda\|w\|^2 + \sum_{x\in D_s, y\in L_s} h(p_y(w) - p_x(w)) \quad (4)$$

Here D_s is the set of point where s is linked to, and L_s is the set of point where s is not linked to. h is the hinge-loss or squared loss function that penalizes violated constrains. It is designed such that when a non-neighbor node is given higher rank than the real neighbor nodes, there will be a penalty, i.e. if $p_y - p_x < 0$ then $h() = 0$ while for $p_y - p_x > 0$, $h() > 0$.

We write p as a function of weight w because the transition of random walk is decided by w. In details, the transition matrix Q can be calculated as:

$$\begin{aligned} a_{uv} &= f_w(\phi_{uv}) \\ Q'_{uv} &= \frac{a_{uv}}{\sum_l a_{uw}} \mathbf{1}_{(u,v)\in E} \\ Q_{uv} &= (1-\alpha)Q'_{uv} + \alpha \mathbf{1}_{v=s} \end{aligned} \quad (5)$$

We have the term $\alpha\mathbf{1}_{v=s}$ because it is the PageRank with restart (personalized PageRank).

Then we can calculate the derivative of F with respect to weight w as:

$$\begin{aligned} \frac{\partial F_s(w)}{\partial w} &= 2\lambda w + \sum_{x,y} \frac{\partial h(p_x - p_y)}{\partial w} \\ \frac{\partial F_s(w)}{\partial w} &= 2\lambda w + \sum_{x,y} \frac{\partial h(\delta_{xy})}{\partial \delta_{xy}} (\frac{\partial p_x}{\partial w} - \frac{\partial p_y}{\partial w}) \end{aligned} \quad (6)$$

Where $\frac{\partial h(\delta_{xy})}{\partial \delta_{xy}}$ is determined by which hinge function we use. We will discuss how to obtain $\frac{\partial p}{\partial w}$ in the following section.

5.2.2 Computation of PageRank vector p and its derivative

The Computation of PageRank vector p seems straight forward. After we get the transition matrix Q, we can iterate to compute the page rank p as:

$$p^T = p^T Q \quad (7)$$

The remaining problem is how to compute the derivative of p w.r.t the weight w. For each step of page rank iteration $p_u = \sum_j p_j Q_{ju}$, we have:

$$\frac{\partial p_u}{\partial w} = \sum_j Q_{ju} \frac{\partial p_j}{\partial w} + p_j \frac{\partial Q_{ju}}{\partial w} \quad (8)$$

And the $\frac{\partial Q_{ju}}{\partial w}$ can be calculated as

$$\frac{\partial Q_{ju}}{\partial w} = (1-\alpha) \frac{\frac{\partial f_w(\phi_{ju})}{\partial w} (\sum_k f_w(\phi_{jk})) - f_w(\phi_{ju}) \sum_k \frac{\partial f_w(\phi_{jk})}{\partial w}}{(\sum_k f_w(\phi_{jk}))^2} \quad (9)$$

And in practice for each training step, after updating the new weight, we can calculate the new page rank p and its derivative by:

- Calculate the new transition matrix Q according to the new weight w
- Initialize the p and its derivative.
- Iteratively calculate the page rank value p using equation(7) until converge. We get a series of p_i, $i = 1, 2, \cdots, t$
- Use the equation(9) and p_i to calculate the final derivative p.

Input a weight, we can get the current page rank and also get its derivative. Then we can use equation(6) to calculate the derivative of loss function and then use it to feed the learning algorithm.

5.2.3 Multi-start SGD

For each node s, we can use the algorithm to learn a optimized weight $w_{opt,s}$ that minimize the $F_s(w)$. However, since our goal is to estimate w that make good predictions across many different nodes s, we need to train a global weight that not only works well for the training graph, but also for the newly arrive nodes. Like paper [3], we extended our model to share weights over all training nodes $s \in S$ by redefining the loss function to be:

$$\begin{aligned} F(w) &= \sum_{s\in S} F_s(w) \\ &= \lambda'\|w\|^2 + \sum_{s\in S} \sum_{x\in D_s, y\in L_s} h(p_y(w) - p_x(w)) \end{aligned} \quad (10)$$

So we can train a global weight w that fits for all the nodes on the training graph. It helps us make prediction for points that never appears in the training graph. Also it makes our model less likely to overfit, which improves the generalization. In the practical training, we used stochastic gradient descent method to update our weight.

5.3. Benchmark Methods in Post Cold-start Stage

5.3.1 Ranking based on internal proximity measurement

We extract a set of proximity features in internal network. Given extracted proximity features, we apply each metric individually to all co-purchase pairs, and produce a ranked list by sorting the scores, from which we take top pairs as co-purchase choice.

5.3.2 Supervised learning using internal network analysis

The drawback of proximity features measurement is we treat and evaluate each metric individually and separately. In this way, the correlation between the features as well as the weight assignment are only coarsely proceeded. In this way, we create a feature list with the proximity features in internal network. We take advantage of different algorithms to perform supervised learning and make prediction about possible co-purchase relationship with the model, including Logistic Regression and linear SVC.

5.4. Ensemble Model of multi-network analysis

We propose a methods to study the impact of cold-start results and external network analysis on post cold-start stage. Ideally, we need to filter out random edges in Amazon network and perform multi-start supervised random walk algorithm over the integration of both internal and external network for link inference. However, due to the complexity in computation of page rank part in the method, it is unrealistic since the size of two hops neighbors of Amazon network expands dramatically compared to the direct neighbors in IMDB.

Therefore, we make a compromise to ensemble the supervised learning results over integration of internal network and external network with the scores in multi-start supervised random walk in cold-start stage. Then the algorithm converts to a ranking problem, in which we choose the top candidates as defined below.

$$r = \text{argmax}_k(\alpha RS_i + \beta CF_i) \tag{11}$$

, in which RS refers score in cold-start stage and CF refers the confidence in binary classification of post cold-start stage.

6. Results

In this section, we show the results for baseline models and our proposed methods.

6.1. Evaluation for Cold-stat results

We mainly compare our proposed methods of multi-start supervised random walk on external networks with the benchmark model, including random network and category inference over internal network. Since our proposed methods needs supervised learning process so we filter part of the external network for experiment and use 90% for training and 10% for testing as showed in Table2.

Common type	Train Nmbr	Test Nmbr
Actor	3771	418
Actress	2258	250
Director	1197	132

Table 2: Division of Training and Testing Set of Multi-start Random Walk Algorithm in Cold-start Stage

Method	Precision	Recall	F1 score
Random Network	0.12%	0.23%	0.16%
Top3 Category Inference	3.15%	5.02%	3.87%
Top5 Category Inference	2.80%	7.44%	4.07%
Actor MSSRW	3.35%	7.68%	4.66%
Actress MSSRW	7.80%	6.75%	7.24%
Director MSSRW	8.59%	6.15%	7.17%
Actor + Actress + Director MSSRW	**10.58%**	**11.18%**	**10.87%**

Table 3: Performance comparison of our proposed methods and benchmark in cold-start stage, in which MSSRW is abbreviated for multi-start supervised random walk over external network

We then compare the results of three proposed methods in cold-start stage as displayed in Table 3.

For better clarification of our multi-start supervised random walk over different external network, we list parameter learning results of the weight of each key features during the process in Table4.

6.2. Evaluation for post cold-start stage

In this section we discuss the potential impact of cold-start results and external network information on prediction in post cold-start stage and compare our proposed methods with benchmark models. The benchmark models includes ranking and supervised learning with internal network features only. Here, internal network features refer to the key features we listed in section 4.2. Our proposed methods is to ensemble the results of cold-start stage with the supervised learning using both internal and external features, and perform ranking based on the weighted scores to get top candidates. We use two classifiers including Logistic Regression(LR) and Linear Support Vector Machine Classifier(lrSVC) for classification and confidence calculation. We choose the cold-start results from union of all direct

Network	Common Neighbors	Jaccard Coefficient	Adamic/Adar	PageRank Score
Actress MSSRW	0.00501	0.000524	0.023232	0.000015
Director MSSRW	0.000331	0.00043	0.007802	0.000056
Actor MSSRW	0.001737	0.000019	0.005969	0.000011

Table 4: Learning weight of key network features during process of multi-start supervised random walk in different external network, in which MSSRW is abbreviated for multi-start supervised random walk over external network

neighbors in actors, actresses and directors external networks. The results of precision, recall and f1-measure in post cold-start stage is listed in Table5. For better illustration, we visualize the result in both cold-start stage and post cold-start stage, and conceive learning curves about precision and recall as visualization results in Figure 6.

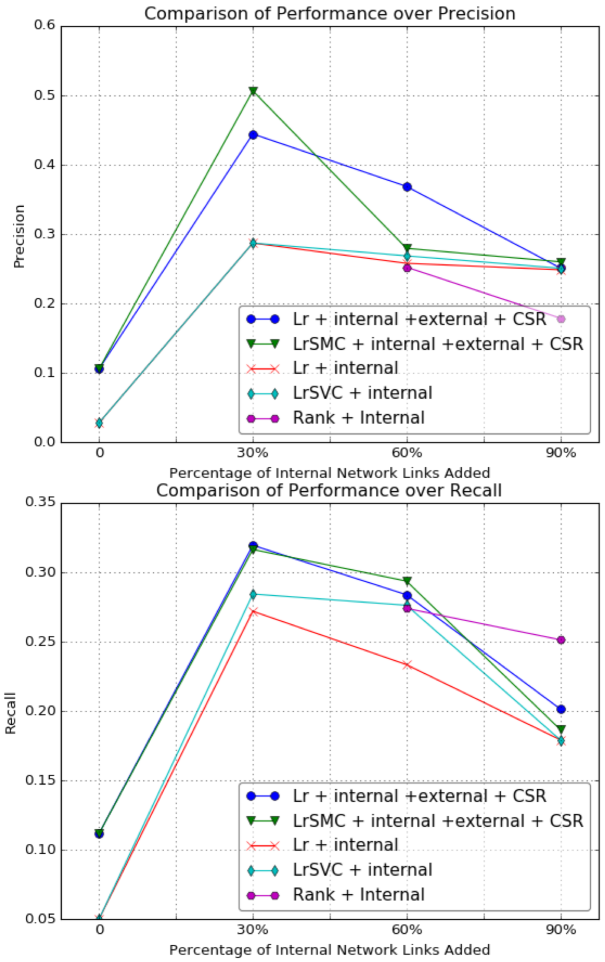

Figure 6: Performance Comparison over Precision and Recall

7. Discussion

In this section, we discuss the results in cold-start stage and early post cold-start stage, and justify why external network features are extremely helpful in internal link prediction in the early stage.

7.1. Discussion in Cold-start Stage

We find that our proposed multi-start supervised random walk algorithm using external network features outperform three benchmark models significantly. Compared to random network model, it raises precision by 88 times and recall by 69 times. Compared to rank methods using category features in internal network, it almost triples the precision and recall.

We find internal network features are less likely to represent the link relationship within same network. Take category as an instance, most of the nodes in internal network would share same categories such as movie or TV show. In this way, part of internal network features tend to be redundant and don't help in link inference and correlation. On the other hand, external information shares good and reasonable correlation with link in internal network and act as complementary resources to mine the patterns in internal links.

Another interesting finding is that the performance of multi-start supervised random walk algorithm on actor external network is not as good as on other two external networks. The reason is that part of actor external network is very sparse. It is observed as the tail part of actor network degree distribution. Therefore, it would be extremely hard to give reasonable inference for that part of network. However, common actress and director serve as an excellent complementary resources to link nodes with sparse link to others. Therefore, the performance of the union of three external network exceeds every single one external network.

7.2. Discussion in Post Cold-start Stage

In visualization of the learning curve with different percentage of internal link added, we observe significant improvement of our proposed methods compared to both ranking and supervised learning using internal information only. When we add only 30% of internal links, our methods outperform supervised learning in benchmark by 21.94% in

Model + feature	Internal Links(%)	Precision	Recall	F1-measure
Rank + Internal	60	25.2309%	27.3985%	26.2701%
	90	17.8733%	25.1326%	20.8903%
Lr + internal	30	28.7016%	27.1917%	27.9262%
	60	25.7762%	23.3378%	24.4965%
	90	24.8529%	17.9215%	20.8256%
LrSVC + internal	30	28.7114%	28.4313%	28.5707%
	60	26.8547%	27.6216%	27.2327%
	90	25.0742%	17.9215%	20.9029%
Lr + internal +external + CSR	30	44.4444%	31.9489%	37.1747%
	60	36.8696%	28.3802%	32.0726%
	90	25.1117%	20.1485%	22.3579%
LrSMC + internal +external + CSR	30	50.6542%	31.6454%	38.9545%
	60	27.9406%	29.3619%	28.6336%
	90	25.9970%	18.6638%	21.7284%

Table 5: The performance comparison of ranking and supervised learning using internal network information only and our proposed ranking of confidence in supervised learning using both external and internal network ensemble with cold-start results

precision.

An interesting case to explain is that the curve reaches peak when 30% internal link added in all cases. It is resulted from imbalance in prediction. Listed in Table 6, when the percentage of internal link increases, we see that the ratio of common edges in candidates deceases dramatically even though more information about internal network is included. It leads to extreme imbalance of network prediction when the percentage of internal links reaches 90%, in which the factor of positive items is around 1.08%. As shown in previous works, link prediction is extremely difficult to solve in occurrence of imbalance[12]. Therefore, the peak of performance reaches when only 30% internal link included, which balances both the information entropy of internal network and data imbalance.

However, we do not provide the results of rank methods when the percentage of included internal link is less than 60%. It is because rank methods based on internal network information is not stable and robust. Ties happen frequently and how to deal with tie impact the performance significantly. When only 30% internal links added, the recall of rank methods would be almost 100% while precision is around 1%. The reason is the lack of diversity in internal network features when with limited links and rank methods cannot distinguish and predict all cases to be positive. It accounts for the fact that at the tail of learning curve, rank methods have worse precision but better recall. It is also the reason why we need to include the results of cold-start, external network features and internal network features in our proposed ranking methods.

The learning curve shows good capability of our proposed model in early stage of inference when there is lack of internal network information. However, with accumulation of internal links, the advantage of our proposed methods shrinks and ultimately have only slight improvement on methods using internal network only. This trend is in accordance with our assumption that we are not to tackle with general link prediction but just the beginning stage.

Intl. LNk%	Common Edges	True Edges	Candidates
30	1252	11442	11714
60	2241	6612	45449
90	943	1718	87209

Table 6: Imbalance in link prediction in post cold-start stage

Moreover, noticing the internal network includes key features such as common neighbor, which is a strong benchmark for comparison in most of link prediction work[2], we believe our contribution and improvement over methods are significant.

8. Conclusion

In this project, we use information in external network to solve cold-start issue of link inference in internal network. We first proposed multi-start supervised random walk algorithm and experiment over external network for link prediction in cold-start stage. In addition, to infer link prediction in post cold-start stage, we come up with ranking methods based on weighted sum scores of cold-start stage results and confidence in supervised learning using both internal and

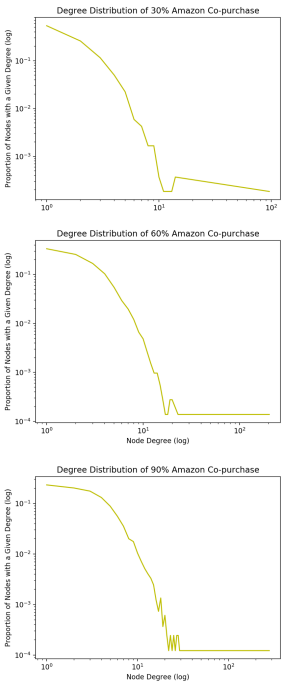

Figure 7: Degree Distribution of External Network Linking Movies with Different Percentage of Internal Link Added

external network features . We prove external network is extremely helpful in link prediction of cold-start stage and early phase of post cold-start stage, with rise of 7.78% and 21.94% in precision in both case.

9. Contribution

All authors contributed equally in the development of the code and the writing of the report.

10. Acknowledgement

The authors would like to thank Prof. Jure Leskovec for his instruction throughout the course, and the TAs for their assistance during this quarter.

References

[1] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM06: workshop on link analysis, counter-terrorism and security*, 2006.

[2] Authors. Frobnication tutorial, 2014. Supplied as additional material `tr.pdf`.

[3] L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 635–644. ACM, 2011.

[4] 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.

[5] H. Kashima, T. Kato, Y. Yamanishi, M. Sugiyama, and K. Tsuda. Link propagation: A fast semi-supervised learning algorithm for link prediction. In *Proceedings of the 2009 SIAM international conference on data mining*, pages 1100–1111. SIAM, 2009.

[6] V. Leroy, B. B. Cambazoglu, and F. Bonchi. Cold start link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 393–402. ACM, 2010.

[7] J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, 1(1):5, 2007.

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

[9] Z. Lu, B. Savas, W. Tang, and I. S. Dhillon. Supervised link prediction using multiple sources. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 923–928. IEEE, 2010.

[10] J. McAuley, R. Pandey, and J. Leskovec. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM, 2015.

[11] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.

[12] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In *Joint european conference on machine learning and knowledge discovery in databases*, pages 437–452. Springer, 2011.

[13] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260. ACM, 2002.

[14] Z. Wang, J. Liang, R. Li, and Y. Qian. An approach to cold-start link prediction: establishing connections between non-topological and topological information. *IEEE Transactions on Knowledge and Data Engineering*, 28(11):2857–2870, 2016.

[15] Y. Zhang and M. Pennacchiotti. Predicting purchase behaviors from social media. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1521–1532. ACM, 2013.