A Link Prediction Framework for Hotel Recommendations

Yiğit Sevim, Günce Keziban Orman, and Orkun Mehmet Kılıçlıoğlu

***Abstract*—Successfully predicting rarely occurring events in large systems can be extremely valuable in various scenarios, such as fraud detection, quality control, sales prediction, etc. In tourism, predicting connections between hotels via their similarity scores can form the basis of a hotel recommendation engine. In this work, we propose a link prediction framework for such an application. This framework first extracts a hotel-to-hotel network from hotel-customer raw data sets. Then, it applies various link prediction approaches. Besides employing well-known node similarity metrics such as Adamic Adar, Jaccard Coefficient, and Preferential Attachment, we also contribute to developing their weighted versions. These six metrics are executed in the basic supervised task of link prediction. The results are evaluated by precision and AUC. In our experiments, we used two novel data sets from the tourism sector: SeturTech and Otelpuan. The results demonstrate that the proposed weighted Adamic Adar returns the most accurate link predictions.**

***Index Terms*—Link Prediction, Smart Tourism, Node Similarity, Precision.**

I. Introduction

Recommendation systems are intelligent mechanisms that explain unseen relations between users and items. Right now, we come across several different approaches from different domains [1]. An interesting and useful application of recommendation systems can be a smart tourism approach that suggests new hotels for travelers when they plan their journeys. In this work, we investigated methods for developing an accurate recommendation engine for smart tourism.

Technically, there are two types of recommendation systems, independent of sector or application type: customized and non-customized.The non-customized system gives the same item recommendation to all system users without considering users' individual interests. In contrast, the customized one considers the preferences or interests of each user, thus recommending certain items to the user more effectively. Indeed, customized approaches are more interesting because of the personalisation of system users.

Most of the traditional approaches use learning methods such as clustering, classification, and regression techniques [2], [3]. They all rely on tabular data, i.e., the attribute-based description of individual objects. These approaches can skip the collaborative aspects that can occur in the systems because of the interactions between the users. We come across efficient methods using complex networks instead of tabular data, thereby allowing one to leverage relational information, i.e., connections between objects. These methods can be counted as collaborative filtering contributions.

Complex networks are advanced graph-based modeling tools for systems, including interacting objects[4], [5]. In this model, the objects are represented by nodes, and their interactions correspond to links between nodes. In the hotel recommendation case, the system corresponds to the e-commerce data sets, including hotel preferences of customers, with the features of hotels and customers. The use of complex networks allows us to consider both the system effects and local effects all at once in the analysis. That is why in this project, our ultimate aim is to generate a new hotel recommendation system for smart tourism through link prediction on this complex network model.

Usually, the recommendation problem is reduced to a link prediction problem. Thus, specifically, we focus on link prediction task for smart tourism. For example, Li et al. [6] consider the recommendation to be a link prediction problem in user–item interaction bi-partite networks. They define a network kernel for the user–item pair's context and use the network topology to infer whether a user may have a link with an item. Another work that uses a link prediction method for bi-partite networks with homogeneous node similarity is described in [7]. The aim was to propose new music to users. Recommendations are mainly based on user-user similarities. In our work, we use bi-partite networks as it was done in the previous works. Network-based recommendation can efficiently use heterogeneous information in networks by expanding neighborhoods and calculating proximity between users and item types [8]. Several link prediction measures are summarized in [9] for various networks.

Our main contributions can be folded into four parts: first, we propose a framework for the link prediction process, tackling to connect the similar hotels in terms of the travelers' preferences. Second, we applied well-known baseline link prediction metrics such as the Jaccard Index, Adamic Adar, or Preferential Attachment [9]. Third, we proposed modifications to the mentioned measures in a way to account for the power of the links. Fourth, we tested our framework on two novel data sets, SeturTech and Otelpuan. Among them, SeturTech is a network from one of the biggest companies in tourism agencies, while Otelpuan is a web site for searching for hotels in Turkey. The rest of the document is organized as follows; in section II we describe the problem we tackle in, explain link prediction methods and also proposed weighted link prediction metrics. Then, in section III, we explain the experiments and related results by interpreting them. Finally, we summarize the work by giving future perspectives in section IV.

Manuscript received March 25, 2022; revised March 31, 2022. This work was supported by the SeturTech R&D Department.

Y. Sevim and O. Kılıçlıoğlu are researchers from SeturTech R&D Department, Istanbul, Turkey (yigit.sevim@setur.com.tr), (orkun.kilicliogl u@setur.com.tr) and G.K. Orman is an Assistant Professor of Computer Engineering Department, Galatasaray University, Istanbul, Turkey, (korman@gsu.edu.tr).

ISBN: 978-988-14049-3-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

II. Method

We propose a framework having several tasks for achieving a proper link prediction for hotel recommendations. In the next parts, we will describe the details of this framework, the link prediction methods that we use, as well as the performance evaluation metrics for the link prediction.

A. Link Prediction Framework

All of the travelers' connections and their hotel choices constitute a complex system. In this system, the hotel preference of a traveler may affect another traveler's choice, even if they do not meet or do not have any connection, because they belong to the same system. The entire system works as one monolithic structure, including small particles, which all have their own local effects as well as system-dependent global effects. Complex networks are one of the most appropriate models for such systems. That is why we focus on complex network modeling for hotel and traveler systems, taking nonlinearity, feedback, and other hidden effects into account.

We propose a framework that models the hotel-traveler data set in the form of an appropriate complex network and finds the proper hotel suggestions. The flowchart of this framework is shown in Fig. 1. The first task of our framework is building a proper network. There are several different types of complex networks, such as simple, attributed, directed, dynamic, or bi-partite. They all have different functionality for different network modeling.

In our case, S is a complex system that includes the travelers' U and the hotels' H. In this system, the travelers' are connected with the hotels that they choose. In its simplest form, S is a user-item system. One of the most appropriate network models for these systems is bi-partite networks. Let us define the bi-partite network as $G_0 = (V_H, V_U, L)$. V_H is the hotel node set whose members are the node representations of the hotels from H. V_U is the customer node set whose members are the node representations of the travelers from U. L is the link set, whose members are the node pairs between V_H and V_U. If a traveler visits or prefers a hotel, there is a link between their represented nodes.

Using bi-partite networks requires a link prediction mechanism for a hotel recommendation problem. This task can be completed directly from the bi-partite network model by identifying appropriate missing links between a hotel type node in V_H and a traveler type node in V_U. However, link prediction techniques for bi-partite networks are case-specific and limited to [10]. That is why we do not directly use bi-partite networks but transform them into uni-partite ones. We come across several link prediction methods based on the idea of forming complete triangles or other local information [11].

We define an auxiliary projected network $G_{test} = (V, L)$ of G_0 as the pair of the V node set and the L link set, with nodes from V representing hotels and links from L representing connections if any two hotels are linked to at least one common user in G_0. Furthermore, from G_0, a weighted version of G_{test} can be formed, with the weight of each link corresponding to the number of common users shared by two hotels in G_0. In this case, the weight of a link shows its strength, i.e. a type of similarity in this specific hotel case. G_{test} has a homogeneous node structure. Standard complex network analysis and link prediction techniques can be easily applied to G_{test}. The second task of our framework is extracting G_{test} from G_0.

In our framework, once G_{test} is extracted, the hotel recommendation is inferred by finding missing links between hotels. We recommend to users hotels that they did not visit yet and that have been linked by prediction process to hotels they have previously visited. Thus, the third and main task of our framework is link prediction. We handle this task as a supervised learning task. G_{train} was created by removing N randomly selected links from G_{test}. On the G_{train} network, we used the link prediction methods described in the following section. The predicted links were evaluated as true or false predictions by determining whether or not G_{test} contained the predicted links. Then the performance of the link prediction methods is calculated for each method. In the next part, we will explain the baseline link prediction methods and our modifications to them, as well as the performance evaluation metric that we use in detail.

B. Link Prediction Methods

Link prediction methods for uni-partite networks are based on two ideas: constituting clusters or constituting triangles. Finding the missing links for the cluster constitution uses network global topological information, whereas the other one uses local information mostly related to node-to-node topological similarity. In this work, we applied both approaches, but in our specific hotel network cases, the clustering was not efficient. The networks that we have built do not have a modular structure. That is why we do not concentrate on clustering-based link prediction here but rather detail triangle-forming methods in detail. The baseline link prediction approaches of this work use pre-defined node-to-node similarity metrics in the supervised learning task. In the following parts, we will explain in detail three well-known metrics: Adamic Adar, Jaccard Similarity, and Preferential Attachment. Our contributions to link prediction are defining case-specific weighted versions of these three similarity metrics. The readers will also find the analytic explanation of the weighted versions in the following part.

Adamic Adar (AA) computes the similarity of nodes based on their shared adjacent nodes. In the equation 1, X and Y are two distinct nodes from V. $N(.)$ is the first-level neighborhood, i.e. the nodes that are directly linked to a node. Thus, AA is the sum of the inverses of the logarithms of the neighbor numbers of each of the common neighbors of two nodes. If X and Y have considerable numbers of common neighbors and those nodes have few connections, then AA score becomes high, i.e. X and Y are *similar*.

$$AA(X,Y) = \sum_{u \in N(X) \cap (Y)} \frac{1}{log(|N(u)|)} \tag{1}$$

We defined Weighted Adamic Adar (WAA) for measuring the similarity of nodes based on the weights of the links of their common neighbors. The definition is given in the equation 2.As a result, instead of the number of neighbors, $N(.)|$, we use the total amount of weight, $W(.)$, that is associated with a node. The WAA differs from AA in a way that if a common neighbor of X and Y has few but powerful

ISBN: 978-988-14049-3-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

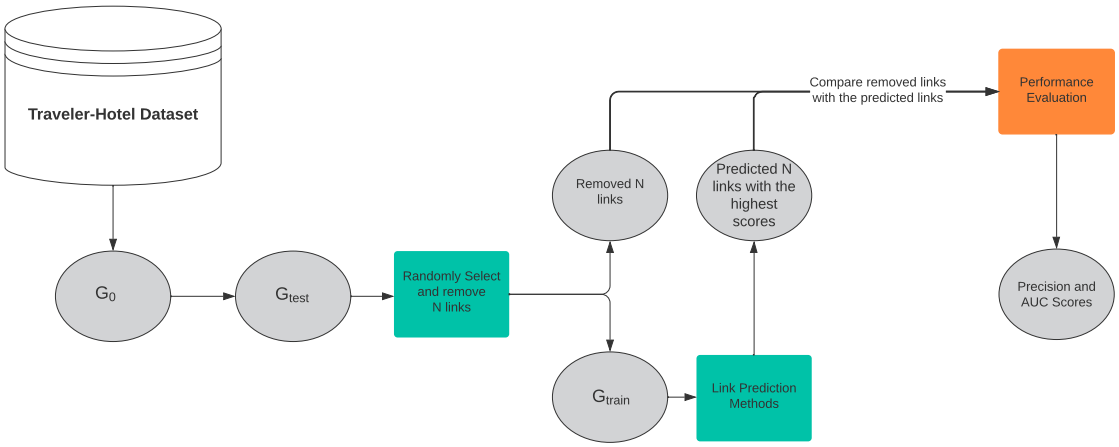

Figure 1. Flowchart of the link prediction framework

connections, which means the weight of its connections is high, the WAA becomes less.

$$WAA(X,Y) = \sum_{u \in N(X) \cap N(Y)} \frac{1}{log(W(u))} \tag{2}$$

Another baseline, the Jaccard coefficient (J), computes the similarity of two nodes by dividing the number of common neighbors of nodes by the number of all neighbors that node pairs have, see equation 3. The more the common neighbor number, the higher the Jaccard similarity.

$$J(X,Y) = \frac{|N(X) \cap N(Y)|}{|N(X) \cup N(Y)|} \tag{3}$$

We defined the weighted version of the Jaccard coefficient (WJ) in equation 4. It is the division of the sum of weights of common neighbors of X and Y by the total sum of weights of all neighbors that node pairs have. The main difference between J and WJ is that J is more sensitive to the number of common neighbors, while WJ is more sensitive to the connection strengths that arelinked to the common neighbors.

$$WJ(X,Y) = \frac{\sum_{u \in N(X) \cap N(Y)} W(u)}{\sum_{u \in N(X) \cup N(Y)} W(u)} \tag{4}$$

The last baseline that we use is Preferential Attachment (PA). Differently from previously explained similarities, PA does not quantify information about common neighbors but directly measures the number of links directly connected to studied nodes (see equation 5). It is computed with the multiplication of the numbers of neighbors of node pairs. It assumes that if a node is more connected, then that node is more likely to have new links.

$$PA(X,Y) = |N(X)| * |N(Y)| \tag{5}$$

Its weighted version (WPA) is given in the equation 6. We use the weight of the links, $W(.)$, rather than the number of directly connected links. We multiply the sum of the weights of the links with the direct neighbors of node pairs.

$$WPA(X,Y) = \sum_{u \in N(X)} W(u) * \sum_{u \in N(Y)} W(u) \tag{6}$$

Higher values for all three baselines and also for their weighted versions indicate a higher probability of connectivity between pairs of nodes. Since all these metrics are not scale-invariant and the associated null model is not generally defined, they do not have any lower bounds, which is the limit of being similar. For precisely this reason, after calculating the similarity of all possible pairs of nodes, we cannot have the possibility of predicting the links by using a threshold or limit value. Due to the analytic nature of these similarities, we will rank the similarities of the node pairs from largest to smallest and suggest links between the most similar ones. It is possible to define a null-model with the system parameters and find statistically significant limits for those metrics, but it is outside the scope of this current work.

C. Performance Evaluation

In our link prediction case, the number of links to be predicted is too small when compared to the total number of possible links in the network. That is why our prediction problem can be seen as similar to the well-known anomaly detection problems. We will find out about rarely occurring events. One of the most suitable performance metrics for these cases is the precision score and AUC [12].

Precision score is being used in information retrieval and classification tasks when the ratio of true positive predictions among positive predictions is important in evaluating the performance. To calculate precision, the number of true positive and false positive predictions has to be calculated. Then, precision score can be calculated by dividing the number of true positive predictions by all positive predictions. The generic definition is given in the equation 7.

$$Precision = \frac{TP}{TP+FP} \tag{7}$$

In the training phase of the experiments, we first remove the randomly selected N links from G_{test} and so, formed G_{train}. Let Λ be the set of removed N links. N links with the highest similarity scores are accepted as true predicted links. We call the set of predicted links according to the similarity metrics $\hat{\Lambda}$. The size of the intersection of Λ and $\hat{\Lambda}$ sets gives us the true positives of the experiment. Furthermore, the size of the set of the elements from $\hat{\Lambda}$ but not from Λ is the

ISBN: 978-988-14049-3-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table I
BI-PARTITE NETWORK PROPERTIES

	SeturTech	Otelpuan
Customer Node Number	45332	170517
Hotel Node Number	1552	959
Link Number	57262	179996
Avg. Deg.	2.44	2.10

false positives of the experiment. Thus, the precision formula becomes the equation 8.

$$Precision = \frac{|\Lambda \cap \hat{\Lambda}|}{|\Lambda \cap \hat{\Lambda}| + |\Lambda \setminus \hat{\Lambda}|} \tag{8}$$

Another commonly used metric for link prediction is the area under the receiver operating characteristic curve, a.k.a. AUC. Originally, AUC is developed for machine learning problems, but its traditional definition is adapted to complex network domain for link prediction. It assesses link prediction performance across the entire list of candidate predictions, in other words, node-to-node similarity of all non-existing links in our case, whereas the precision just considers the first candidate links with the highest ranks or scores. In equation 9, AUC formula is given.

$$AUC = \frac{n^{'} + 0.5n^{''}}{n} \tag{9}$$

Here, n is the number of all comparisons between the true missing links and all predicted links except missing ones, $n^{'}$ is the number of cases where the true missing links have a higher similarity scores, and $n^{''}$ is the number of cases where the true missing links have an equal similarity score.

III. EXPERIMENTS AND RESULTS

A. Data Sets

We have used two distinct data sets in this study. The first data set contains historical hotel sales in Setur between 2013 and 2019. Setur Servis Turistik AS provides travel bookings for air, land, and sea travel for both individuals and businesses. Setur also provides services for duty-free goods, and Setur is one of the leading tourism agencies in Turkey. The data set was provided by SeturTech R&D department and has columns for customer and hotel id's, dates of purchase and entry into the hotel, hotel features such as location, services, and customer features such as age, gender, etc.

Second data set was collected by SeturTech R&D department from Otelpuan.com website by using web scraping methods. The data set contains the customer ratings in the 1–10 range for hotels on the Otelpuan.com website. Otelpuan was founded in 2008 to inform travelers about tourism services by collecting ratings and comments on their website. We first extract bi-partite networks from these two raw datasets. The statistics related to bi-partite networks are listed in table I.

Here, we listed the numbers of all nodes, although there are a large portion of not linked hotel nodes in SeturTech. There are more customers with fewer hotels in Otelpuan. Its total link count is considerably higher than the SeturTech network. We compute the average degree as the average number of links connected to hotel-type nodes. Two networks have similar average degrees.

Table II
SETURTECH PROJECTED NETWORK PROPERTIES

	Network	Largest Component
Node Number	1552	1351
Link Number	14378	14359
Average Path Length	2.78	2.78
Transitivity	0.22	0.22
Average Degree	20.97	21.25
Diameter	7	7
Density	0.0153	0.0157
Degree Centralization	0.23	0.24
Betweenness Centralization	0.0747	0.0769
Closeness Centralization	0	0.36
Eigenvector Centralization	0.91	0.91

B. Network Topological Properties

Projected networks are extracted from bi-partite ones as it is explained in section II. We examine the topology of extracted projected networks. The values of the well-known topological features for SeturTech are listed in table II. Although the network has more than 10000 hotel nodes, most of them have no connection, neither in bi-partite nor in the projected network. There are a small number of small components. Except for those small components, the rest of the network is one connected structure. That is why interpreting the largest component's topology instead of the entire network's one is more meaningful for SeturTech.

Accordingly, both the diameter and transitivity of SeturTech's projected network are compatible with well-known complex networks [4]. But its density is too high which might make to distinguish meaningful substructures difficult in the network. A similar topological analysis is applied to the Otelpuan projected network. The values of its topological features can be found in table III. It seems Otepuan is even denser than SeturTech network.

We also examined if there might be any community structure in both networks because the existence of the communities affects the link prediction. We applied well-known partitioning community detection methods: Leading Eigenvector, Walktrap, FastGreedy, Louvain, Infomap and Label Propagation. The details of those algorithms can be found in [13]. The algorithms' performances differ from each other since they find a different number of communities at different sizes. Furthermore, their modularity scores are also different. Since no algorithm finds highly modular community structures, the networks have no significant clusters. That is why we do not do cluster-based link prediction results.

Considering these two networks together, we infer that in Turkey, the hotels are not segmented according to the travelers' preferences. The SeturTech system seems to have highly preferred hotels. Its network's eigenvector centralization was high. The hotels with the highest centrality can be the most popular hotels in this system. Otelpuan does not seem to have such central nodes. Because the hotel systems are not clustered, it can be more meaningful to concentrate on local topological information and hotel-related features. We do not consider hotel features yet, but in the next part, we will explain in detail the results of node similarity-based link prediction.

ISBN: 978-988-14049-3-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table III
OTELPUAN PROJECTED NETWORK PROPERTIES

	Network	Largest Component
Node Number	959	805
Link Number	14008	14005
Average Path Length	2.61	2.61
Transitivity	0.67	0.67
Average Degree	29.21	34.79
Diameter	6	6
Density	0.03	0.04
Degree Centralization	0.22	0.26
Betweenness Centralization	0.03	0.04
Closeness Centralization	0	0.35
Eigenvector Centralization	0.84	0.81

C. Link Prediction Performances

We have observed the link prediction results as precision and AUC values for different experiment scenarios. As common steps for each experiment scenario, we have applied different steps of our framework, which are explained in section II. To begin, we have designated the main bipartite network as G_0. Secondly, we have formed G_{test}, the projected network of the hotels. In the third step, we removed N randomly selected links from G_{test} and formed G_{train} respectively. The values of N were selected from the set of $\{1, 10, 20, 50, 70, 100, 200, 500, 1000, 5000, 10000$ and 10% of the number of links for SeturTech and Otelpuan G_{test} networks$\}$ consecutively. In the experiments, we not only aimed to compare different similarity scores' performances in different data sets but also aimed to observe their sensitiveness to the number of removal links. All link prediction methods described in the subsection II-B were applied to the G_{train} for each N. Statistical significance was ensured by repeating all experiments five times. The average precision scores of five iterations when $N <= 100$ as well as when N is the 10% of the number of links are shown in the Table IV and Table VI for SeturTech and Otelpuan respectively.

Table IV
SETURTECH LINK PREDICTION PRECISION SCORES

	1	5	10	20	70	100	1437
AA	0.0	0.04	0.02	0.04	0.062	0.07	0.216
J	0.0	0.00	0.00	0.02	0.017	0.03	0.152
PA	0.0	0.04	0.06	0.05	0.034	0.066	0.163
WAA	0.0	0.04	0.04	0.05	0.082	0.06	**0.223**
WJ	0.2	0.04	0.06	0.02	0.065	0.066	0.205
WPA	0.0	0.04	0.02	0.02	0.034	0.05	0.169

In the table, we show the results for the limited N values for the sake of readability. However, in figure 2, the precision scores for SeturTech networks as a function of all N are shown. Accordingly, For all $N <= 50$, average precision scores are low. For $N > 50$, precision values have increased until we increased N to 5000. Highest precision score is 0.223 and it was obtained with using the WAA we have proposed in subsection II-B for the N value of 1437 (%10 of the total links).

A remarkable fact about the SeturTech results is that the rate of finding true links is too low when we look for a small number of links. Indeed, this can be an expected fact. Complex networks have a sparse structure because of the large number of unseen links. It causes there to be too many possible links to add, in other words there is a too low probability of finding the correct links. Thus,

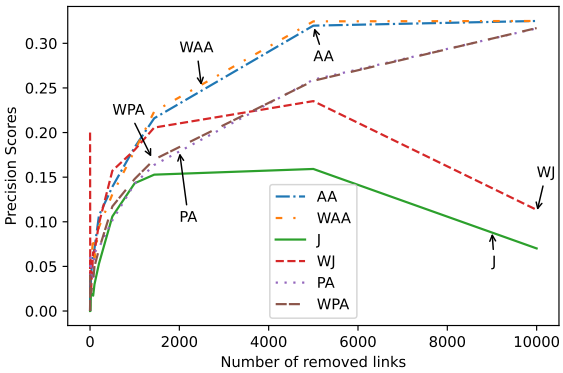

Figure 2. SeturTech Networks' precision scores for different numbers of removed links

the higher the number of links to predict, the higher the precision, independently of the studied similarity metric. However, we observe the opposite behavior at the J and WJ when $N > 5000$. Their precision has decreased while the rest of the similarity metrics have increased their performance. This can be due to the fact that SeturTech seems to have a centralized structure (high eigenvector centrality) rather than a local structure (low transitivity). Removing many links causes several connected components to break out. Because there are not many triangle connections, nodes lose their common neighbors. This results in a decrease in Jaccard-based similarities.

Table V
SETURTECH LINK PREDICTION AUC SCORES

	1	5	10	20	70	100	1437
AA	0.918	0.992	0.880	0.872	0.967	0.926	**0.933**
J	0.931	**0.994**	0.875	0.849	0.951	0.905	0.908
PA	0.916	0.925	0.904	0.902	0.881	0.925	0.916
WAA	0.913	0.993	0.880	0.873	0.968	0.927	**0.933**
WJ	0.981	0.992	0.867	0.858	0.954	0.907	0.912
WPA	0.919	0.924	0.898	0.910	0.878	0.925	0.916

AUC results (see table V), also supports the same facts that we obtained with precision results. The highest score for all experiments is obtained for J when the number of removed links is 5. Differently from precision, removing higher numbers of links seems to not affect too much the performance of different metrics. Comparing the scores for the case of removing %10 of the links, AA and WAA outperform other similarity metrics. Regarding the experiments done with Otelpuan data, for all link prediction methods except WJ, after the small fluctuations on initial N values, precision scores have increased for all the N values. The precision performances are much more higher than the one of SeturTech. WA performance have been decreased for the $N > 5000$. Highest precision score of 0.68 was achieved with both AA and WAA methods we have for the N value of 5.

As with the SeturTech case, the response of the various similarity metrics to an increase in N is not identical to that of Otelpuan. Except J and WJ, the precision scores increase up to a limit, then they stay straight. But for these two metrics, although it is not as significant as in the case of SeturTech, they still exhibit a decrease when removing more than 5000 links. According to its topological structure, Otelpuan has

ISBN: 978-988-14049-3-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table VI
OTELPUAN LINK PREDICTION PRECISION SCORES

	1	5	10	20	70	100	1400
AA	0.6	**0.68**	0.6	0.67	0.625	0.584	**0.627**
J	0	0	0	0.2	0.587	0.594	0.618
PA	0.2	0.16	0.26	0.39	0.482	0.52	0.619
WAA	0.4	**0.68**	0.46	0.59	0.588	0.618	**0.627**
WJ	0	0	0	0.21	0.545	0.636	0.616
WPA	0.2	0.36	0.28	0.33	0.451	0.498	**0.627**

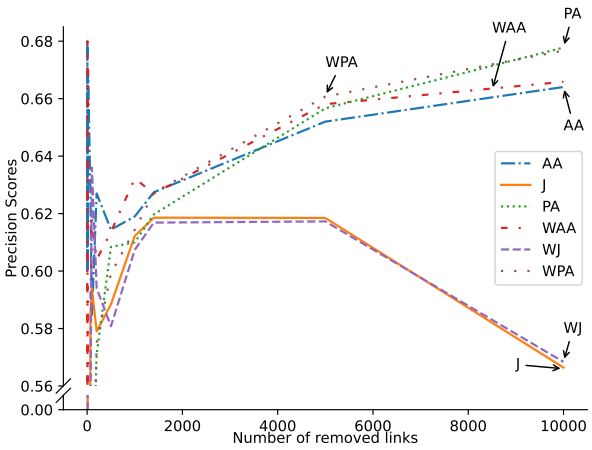

Figure 3. Otelpuan Networks' precision scores for different numbers of removed links

high transitivity. Thus, it seems to have more local structures than SeturTech. This can result in a decrease in Jaccard-based similarities. We show the AUC scores for Otelpuan in table VII. This result supports our previous comments as well. When the number of removed links is small, we obtain even higher scores. For instance, WPA gives the highest results when we remove %10 of the links. The weighted versions of the similarity metrics outperform their traditional forms for both metrics and both networks.

Table VII
OTELPUAN LINK PREDICTION AUC SCORES

	1	5	10	20	70	100	1400
AA	1.0	0.998	0.970	0.986	0.963	0.919	0.940
J	1.0	0.998	0.956	0.985	0.953	0.908	0.933
PA	1.0	0.997	0.954	0.965	0.956	0.927	**0.955**
WAA	1.0	0.998	0.970	0.986	0.963	0.919	0.940
WJ	1.0	0.998	0.956	0.985	0.953	0.908	0.934
WPA	1.0	0.997	0.954	0.965	0.956	0.927	**0.956**

Overall, experiments for each data set show that the higher the number of removed links, the more performing the link prediction in general. Moreover, our proposed weighted metrics outperform their unweighted versions in almost all cases. Among the similarity metrics, J and WJ seem to be the most sensitive ones to the network structure. Among different systems, we obtained higher scores for Otelpuan which is denser and has higher transitivity. In this work, we used the local node-to-node similarity metrics. Thus, we infer that predicting the links of a network that has local structures via these measures is easier. Among all metrics, the performance of AA and WAA is one step ahead of the rest. Their scores did not decrease even when we removed the majority of the links. These results give us an insight into the strength of the metrics.

IV. CONCLUSION

In this work, we concentrate on complex network modeling for smart tourism. More specifically, we address the link prediction problem on hotel-to-hotel networks. We proposed a framework for extracting hotel-to-hotel networks from hotel-customer raw data sets and evaluated different link prediction methods. We used both well-known local approaches and proposed weighted versions for two real-world systems, SeturTech and Otelpuan. The topological features of the extracted network showed that both systems have realistic properties. The link prediction performances highlight that our proposed weighted Adamic Adar is the most suitable metric for an accurate link prediction on those systems.

Different perspectives on this work can be first, taking into account hotel attributes and different network-based similarity metrics, and second, using machine learning approaches for link prediction instead of basic ranking. In this work, as the first step of building a recommendation engine, we mainly concentrate on the link prediction task. We did not produce business-type hotel recommendations. A supplementary step of transforming analytic results into business cases can also be an interesting perspective.

REFERENCES

[1] A. C. C, *Recommender systems*. Cham: Springer International Publishing, vol. 1.
[2] S. Seyednezhad, K. Cozart, J. Bowllan, and A. Smith, a review on recommendation systems: Context-aware to social-based. arXiv preprint arXiv:1811.11866.
[3] G. W. W. and L. F, "Research on collaborative filtering personalized recommendation algorithm based on deep learning optimization," in *2019 International Conference on Robots Intelligent System (ICRIS*. IEEE, p. 90–93.
[4] M. E. J. Newman, "The structure and function of complex networks," *SIAM Rev.*, vol. 45, no. 2, p. 167–256, jan 2003. [Online]. Available: https://doi.org/10.1137/S003614450342480
[5] M. Mitchell, "Complex systems: Network thinking," *Artificial Intelligence*, vol. 170, no. 18, pp. 1194–1212, 2006, special Review Issue. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S000437020600083X
[6] L. X. and C. H, "Recommendation as link prediction in bipartite graphs: A graph kernel based machine learning approach," *Decision Support Systems*, vol. 54, no. 2, p. 880–890.
[7] Z. L., Z. M., and Z. D, "Bipartite graph link prediction method with homogeneous nodes similarity for music recommendation," *Multimedia Tools and Applications*, pp. 1–19,.
[8] S. B. and H. S, "Graph-based collaborative ranking," *Expert Syst. Appl*, vol. 67, no. C, p. 59–70.
[9] L. T. J. and B. S.D, "Link prediction measures in various types of information networks: a review," in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM*. IEEE, p. 1160–1167.
[10] J. Kunegis, E. W. D. Luca, and S. Albayrak, "The link prediction problem in bipartite networks," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*. Springer, 2010, pp. 380–389.
[11] E. C. Mutlu, T. Oghaz, A. Rajabi, and I. Garibay, "Review on learning and extracting graph features for link prediction," *Machine Learning and Knowledge Extraction*, vol. 2, no. 4, pp. 672–704, 2020.
[12] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Statistical Mechanics and Its Applications*, vol. 390, no. 6, pp. 1150–1170, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S037843711000991X
[13] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0370157309002841

ISBN: 978-988-14049-3-0
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)