Learning-based link prediction analysis for Facebook100 network

Tim Poštuvan[a,b,1] **and Semir Salkić**[a,2]

[a]University of Ljubljana, Faculty of Computer and Information Science, Ljubljana, Slovenia; [b]University of Ljubljana, Faculty of Mathematics and Physics, Ljubljana, Slovenia

In social network science, Facebook is one of the most interesting and widely used social networks and media platforms. In the previous decade Facebook data contributed to significant evolution of social network research. Paired with this topic we have experienced growing popularity in the link prediction techniques, which are important tools in link mining and analysis. This paper gives a comprehensive overview of link prediction analysis on the Facebook100 networks, which were derived in 2005. We study performance and evaluate multiple machine learning algorithms on this network. We use networks embeddings and topology-based techniques such as node2vec and vectors of similarity metrics. Using these techniques similarity features for our classification models are derived. Further we discuss our approach and present results. Lastly, we compare and review our models, where overall performance and classification rates are presented.

Link prediction | Social networks | Classification | Supervised learning

Social networks became an important focus in our research. We are witnessing exponential user expansion on social platforms (e.g. Facebook, Twitter and LinkedIn). People are joining these platforms and generating substantial amounts of data which can reveal interesting clues about user behaviour, society and psychology. We can see significant increase in research on the topic of social networks and link prediction. In the last decade we are experiencing a rise in this overlapping topic of network science and data science, which is primarily used to analyze and understand social networks (1). According to Facebook estimations in March 2020, they have 1.73 billion DAU (Daily Active Users). Much work has been done to understand complexities and challenges of social networks, where considerable knowledge has been obtained (2, 3) . In that manner we are using the Facebook100 dataset (2005), which includes a complete set of people from Facebook networks for 100 different colleges and universities in the USA. We are using it to present our analysis and discoveries. This paper is a comprehensive detailed overview of link prediction techniques applied to this dataset. We are using a learning-based approach with local probabilistic models such as logistic regression. Besides that we are using standardized link prediction classification models such as: random forests, artificial neural networks and one kernel based models (e.g. SVM). Having in mind that we have two main approaches to link prediction, we chose learning-based approaches instead of similarity-based approaches because we want to compare performance, stability and classification accuracy of multiple classification models instead of showcasing measures for analysis of node proximity. We are using standardized similarity based methods to derive notable features, which are used as model input. The presented methods are node2vec graph embedding and an ensemble of topology-based metrics such as Jaccard Coefficient, Adamic Adar Coefficient and Preferential Attachment Index. Using

these methods we build our classification models which are presented, focusing on performance and generalization, with appropriate discussion and results.

In related work we are presenting previous work and discoveries in the field of link prediction on social networks. In datasets we are giving description and specification of our datasets, which are used in training and testing. In methods we are providing an overview of our implementation plan, where we are stating two main focuses: feature selection and model hyperparameters. In results we are giving an overview of achieved results, while providing comprehensive model analysis. In discussion we are presenting useful insights of results, which are used to explain the interesting nature of used data.

Related work

Link prediction has recently become very popular for prediction of future relationships between individuals of social networks. Consequently, a great variety of different approaches were invented. In the past decade, many efforts have been made by psychologists, computer scientists, physicists and economists to solve the link prediction problem in social networks. According to Wang et al. (1) there are two ways to predict links: similarity-based approaches and learning-based approaches. Similarity-based approaches calculate a similarity score for every pair of nodes, where higher score means higher probability that the corresponding nodes will be connected in the future.

Learning-based approaches are treating the link prediction problem as a binary classification task (4). Therefore, typical machine learning models can be employed for solving the problem. These include classifiers like random forest (5), multilayer perceptron or support vector machine (SVM) (6), as well as probabilistic models. The learning-based approaches use non–connected pairs of nodes as instances with features describing nodes and the class label. Pairs of nodes which have potential to become connected are labeled as positive and the others as negative.

Their feature set consists of similarity features from the similarity-based approaches and features derived from domain knowledge (e.g. textual information about members of social networks). Using combination of both can remarkably improve the link prediction performance. Scellato et al. (7) considered social features, place features and global features in location-based social networks for link prediction based on a supervised learning framework.

Both types of approaches rely on various metrics, which use information of nodes, topology of network and social theory to

[1] tim.postuvan@gmail.com [2] ss9343@student.uni-lj.si

calculate similarity between a pair of nodes. Metrics consist of three categories: node-based, topology-based and social theory based metrics.

Node-based metrics use the attributes and actions of individuals to assess similarity of node pairs. They are very useful in link prediction; however, it is usually hard to get the data because of privacy issues.

Most metrics are based on the topological information and are called topology-based metrics. They are most commonly used for prediction, because they are generic and domain independent. Topology-based metrics are further divided into the following subcategories: neighbor-based, path-based and random walk based metrics. Neighbour based metrics assume that people tend to form new relationships with people that are closer to them. The most famous are Common Neighbors (Newman [8]), Jaccard Coefficient (Salton & McGill [9]), Adamic Adar Coefficient (Adamic & Adar [10]) and Preferential Attachment Index (Barabási et al. [11]). The first three all use the same idea that two nodes are more likely to be connected if they share a lot of common neighbours. On the other hand Preferential Attachment Index assumes that nodes with higher degree have higher probability of forming new edges.

Neighbor-based metrics capture local neighbourhood but do not consider how nodes are reachable from one another. Path-based metrics incorporate this information by considering paths between nodes. They are more suitable to small networks and are not scalable to big networks. Examples of path-based metrics are Local Path (Lü et al. [12]) and Katz metric (Katz [13]). Local Path metric makes use of information of local paths with length two and three, while giving more importance to the paths of length two. Katz metric calculates the similarity by summing all the paths connecting the two nodes, giving higher weight to shorter paths.

Social interactions between members of social networks can also be modeled by random walk, which uses transition probabilities from a node to its neighbors to denote the destination of a random walker from the current node. Examples of random walk based metrics are Hitting Time and SimRank (Jeh and Widom [14]). Hitting time metric calculates similarity based on the expected number of steps required for a random walk starting at a node to reach the other node. SimRank metric computes similarity according to the assumption that two nodes are alike if they are connected to structurally similar nodes.

Social theory based metrics take advantage of classical social theories, such as community, triadic closure, strong and weak ties and homophily, improving performance by capturing additional social interaction information. Liu et al. [15] proposed a link prediction model based on weak ties and degree, closeness and betweenness node centralities.

When designing a feature set, the choice of features tremendously influence the performance of link prediction. Sometimes is it hard to find appropriate features, hence it is desirable that an algorithm learns important features on its own. Network embedding methods aim at learning low-dimensional latent representation of nodes in a network. Embeddings should follow the principle that similar nodes in the network have similar embedding representations. The advantage of node embedding as a technique is enormous since it does not require feature engineering by domain experts. Network embeddings

methods can be broadly categorized into four classes: methods based on random walks, matrix factorization, neural networks, and probabilistic approaches. For the purpose of this paper methods based on random walks are the most relevant.

Methods based on random walks determine similarities using random walks on the original network. The Skip–Gram model, described in Mikolov et al. [16], is then usually used to generate node embeddings from the random walks. Examples of such methods are DeepWalk (Perozzi et al. [17]) and node2vec (Grover & Leskovec [18]). DeepWalk was the first technique for network embeddings, inspired by deep learning. It uses random walks with fixed transition probabilities to measure node similarity, while embeddings are derived using the Skip-Gram model. Node2vec is a generalization of DeepWalk which uses supervised random walks for node neighbourhood exploration. The random walk is controlled by a return parameter p and an in-out parameter q. Then similarly Skip–Gram model is used, but this time approximated via negative sampling, for embedding generation.

Evaluation of the methods play the crucial role in machine learning task in general. To estimate performance of the link prediction approaches more evaluation criteria exists. While some papers utilize $Precision@N_p$ for a range of N_p values (Zhang et al. [19]), others use AUROC (Grover & Leskovec [18]).

Data

We study the Facebook social network of friendships at one hundred American colleges and universities at a single moment of time in September 2005 ([20], [21]). Besides the information about friendships, network also contains limited demographic information. The following information is available for each user: student/faculty status flag, gender, major, second major/minor (if applicable), dorm/house, year and high school. Network is unweighted and undirected. The whole network consists of 3.2 million nodes with 23.7 million links between them [22]. Maximum degree of a single node is approximately 4900 and minimum degree is only 1, with an average of 15. According to statistics network appears to be disassortative but this is only the consequence of its size. It also has high average clustering coefficient 0.097, which is characteristic of social networks.

Dataset	n	m	d	C
Train	4943.5	206247.6	77.26787	0.2808
Unseen	3517.8	140793.2	80.3072	0.2689

Table 1. Structural information of normal and unseen data

Table 1 contains structural measurements for used network sets. All of the presented measurements are average over all networks in the corresponding set. The presented values are: average clustering coefficient (**C**), average degree (**d**), average number of nodes (**n**) and average number of edges (**m**). We can see by number of nodes and edges that train set is larger than unseed dataset. Clustering coefficient and average degree are high, which is one of expected characteristics of social networks.
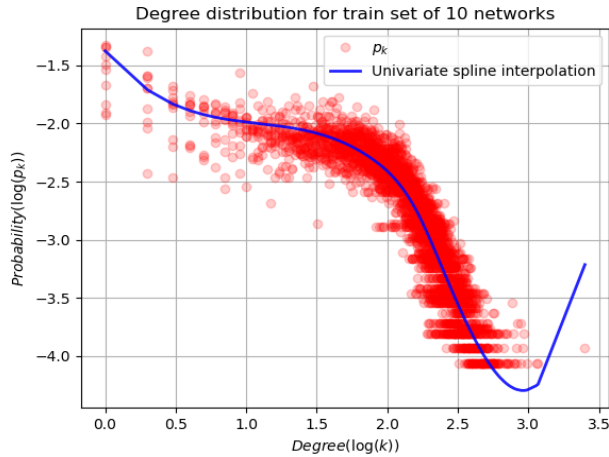
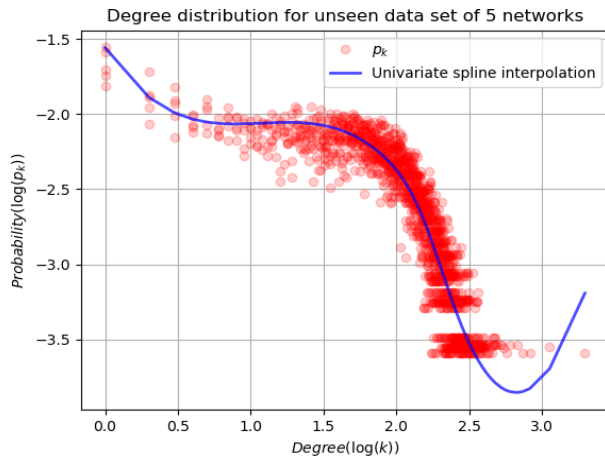**Fig. 1.** Degree distribution of train set

Fig. 2. Degree distribution of unseen data set

Degree distribution for used networks is presented in figures 1 and 2. Degree distribution is presented on a log-log scale for all networks in both sets respectively. For cleaner overview, we used interpolation (Univariate spline) to showcase distribution of all networks. It is visible on both figures that all networks follow power law, which is expected for social networks. Having in mind that we are using real life social networks, it can be concluded that they are scale-free networks by degree distribution results. However, it is interesting to point out existence of big hubs, nodes with very high degree. They are visible on right side of the distribution graph. This is one of the reasons why interpolation at the end of the plot has an unexpected minimum.

Feature set

Feature engineering probably plays the most important role when coping with a machine learning problem. Informative features crucially effect model accuracy, hence the process of feature engineering is usually very time consuming. In learning-based link prediction each pair of nodes is described using a combination of node-based, topology-based and node

embedding features, depending on approach. In this paper we are using three different datasets.

Node-based features. Node-based features use domain-specific information about individuals. Facebook100 dataset has already a basic set of features, however, not all of them are useful for link prediction task. Almost all features had to be transformed, in order to describe node pairs, instead of individuals. From some features, for example dormatory information, new features had to created, because otherwise model would not be transferable between networks. Problem arises from the fact that different universities use different numerations of their dormatories. Considering the above constraints, we derived the following features:

- **is_dorm**: binary value, indicating whether the nodes live in the same dormatory

- **is_year**: binary value, indicating if the nodes started college in the same year

- **year_diff**: numerical value, stating the absolute difference between the years, when the nodes stated college

- **from_high_school, to_high_school**: numerical values, stating indices of nodes' high schools

- **from_major, to_major**: numerical values, stating indices of nodes' majors

- **is_faculty**: binary value, indicating whether the nodes have the same faculty status

- **is_gender**: binary value, indicating if the nodes have the same gender

Since networks are undirected, each pair of nodes must be uniquely represented using above features. Representation should not depend on order of the pair, thus from_major and to_major are ordered in a way that the value of from_major is not greater than the value of to_major. The same holds for from_high_school and to_high_school.

Like the majority of datasets Facebook100 does not contain all information about all individuals. Therefore, missing values had to be handled. We decided that imputing is reasonable only for attribute years, where missing values were substituted with the mean. Values of other attributes were left intact but as soon as one of the nodes in the pair had a missing value, the corresponding binary values was automatically zero.

Topology-based features. The most commonly used features for link prediction are topology-based features. They are particularly useful, when you do not have any problem specific information, because they are generic and domain independent. Although Facebook100 dataset has additional domain specific data, topology-based features still have great impact on model accuracy. In this paper we are using the following topology-based features:

- **Jaccard Coefficient** (9). Jaccard Coefficient normalizes the size of common neighbors. According to Jaccard Coefficient a pair of nodes is assigned a higher value when the nodes share a higher proportion of common neighbors relative to total number of their neighbours.

$$JC(x,y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

where $\Gamma(x)$ is a set of neighbours of node x.

- **Adamic Adar Coefficient** (10). Adamic Adar Coefficient measure is closely related to Jaccard Coefficient. It is calculated as a weighted sum of common neighbours, where common neighbours with fewer neighbours have greater impact. The rationale behind it is that high degree nodes are more likely to occur in common neighbourhood, thus they should contribute less than low degree nodes.

$$AA(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log |\Gamma(z)|}$$

- **Preferential Attachment Index** (11). The measure is based on the concept that nodes with higher degree have higher probability of forming new edges.

$$PA(x,y) = |\Gamma(x)||\Gamma(y)|$$

- **Resource Allocation Index** (23). Resource Allocation Index metric is very similar to Adamic Adar Index. The only difference is that Resource Allocation Index punishes high degree nodes more.

$$RAI(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{|\Gamma(z)|}$$

Node embedding features. Network embeddings methods aim to learn low-dimensional latent representation of the nodes in a network. Generating a dataset comprising of every node in a network we are able to use these representations as features. This can be used for a wide variety of tasks such as classification, clustering, link prediction, and visualization. Using node2vec (18) we were able to generate our embeddings dataset.

The key point is that node2vec is based on random node walks performed in a biased manner across the network. With this generic approach we are able to sample any network in a search for vector representation of its structural properties. With the introduction of search bias α we are able to control our search in BFS or DFS manner. If we choose *"in-out parameter"* (q), walks are more biased to visit nodes further from the start node, thus expressing the nature of exploration. Fixing *"return parameter"* ($p = 1$) ensures that we are less likely to visit same node twice, which in return adopts the strategy of modern exploration (avoids 2-hop redundancy in sampling).

As stated in the case study by Grover & Leskovec (18) for social structures it is beneficial to tune node2vec hyperparameters to discover communities of nodes which are interacting with each other. Capturing this type of behavior using embedding representation is significantly beneficial for the link prediction task. Dimensions were discovered heuristically. Having in mind that we are sampling different networks, the vector dimension should be as small as possible while carrying all relevant information. Using exploration with biased random walk, hyperparameters were arbitrarily chosen to achieve best model performance. We defined our selection of hyperparameters by

this reasoning, which was used to define the starting point for grid search. The final parameters were: 64 dimensions, 50 walks per node, $q = 0.8$ and 20 nodes in each walk. Since base node2vec approach yields embeddings for nodes, we used Hadamard product to express vector representations for edges.

Datasets

Because Facebook100 dataset is enormous, lack of computing power prevented us from considering the complete dataset for analysis. Therefore, we have decided that we would perform analysis only on a subset of networks. We selected ten networks as normal data and five networks as unseen data. Normal data was used for training and testing, whereas using unseen data we evaluated if our models are transferable to new data.

Firstly, we had to preprocess graphs and obtain train and test node pairs. For every graph we used the standard approach of generating an incomplete train graph $G_{train} = (V, E_{train})$ from the original graph $G = (V, E)$. The connected node pairs $\{i, j\} \in E \setminus E_{train}$, which are present in the whole graph but not included in the train graph, are used as positive instances for link prediction task. Positive instances were randomly sampled from the original graph's edge set E. We decided to sample 2% of edges in original graph G. Since dataset should contain positive as well as negative instances, we had to obtain also negative instances – pairs of nodes that are not connected by an edge. These were obtained by using randomly selected node pairs $\{i, j\} \notin E$, which are not in the original graph's edge set. To get a balanced dataset the number of negative instances is the same as the number of positive ones.

In our experiments, all unseen data instances were used for testing models' ability to adapt to new graphs. However, normal data was further split into train and test data. We used standard division: 80% of it was used as the train data and the remaining 20% was used as the test data, both containing approximately the same number of positive and negative instances.

Using this data three datasets were created: baseline, topological and embedding dataset. Each dataset represents node pairs using a different combination of features. Baseline dataset is the simplest one and contains only topology-based features. A bit more complex is topological dataset, which in addition to the topology-based features makes use of node-based features as well. Node pairs in embedding dataset are described using node-based features and Hadamard product of the corresponding nodes' embeddings.

Feature selection

Contemporary datasets usually have abundance of data, which is not always relevant to the problem. Hence, datasets should be preprocessed before models are used on them. Preprocessing takes place mainly to reduce the size of the dataset and achieve more efficient analysis, as well as removing redundant features, which have negative impact on the performance of the model. The aim of feature selection is to maximize relevance and minimize redundancy of the features.

Our feature sets are not enormous, thus feature selection was done solely for the sake of performance improvement of the models. We are using recursive feature elimination with cross-validated selection (RFECV) in combination with linear kernel support vector machine (SVM) to get reduced feature
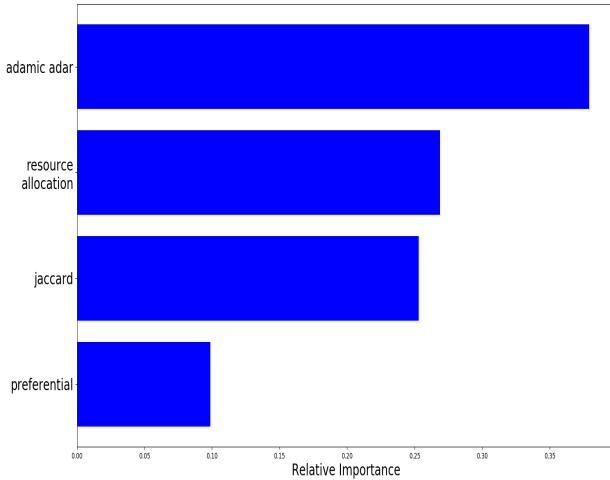
Fig. 4. Feature importance of baseline dataset according to random forest classifier

sets. This method recursively considers smaller and smaller sets of features, while after every iteration prunes the least important features according to the chosen model. It belongs to wrapper methods for feature selection, since it appraises subsets of features based on performance of the modelling algorithm. According to Jović et al. (24) wrapper methods have been empirically proven to yield better results than other methods because subsets are evaluated using real models.

Baseline dataset. The above feature selection method recognized Adamic Adar Coefficient, Jaccard Coefficient and Resource Allocation Index as the most informative features. The most relevant feature is Adamic Adar Coefficient and the least relevant one is Preferential Attachment Index. This is completely coherent with random forest feature importance shown

in figure 4. Adamic Adar Coefficient is the most relevant feature, although Jaccard Coefficient has higher correlation with labels. All selected features are highly correlated with label, whereas Preferential Attachment Index is not. This is probably the reason why Preferential Attachment Index is the only feature which was not selected. From correlation matrix it is also evident that Adamic Adar Coefficient and Resource Allocation Index are almost perfectly correlated, which is expected because of the similarity in their definitions. Nonetheless adding it results in a slightly better performance, thus the algorithm decides to keep it.

Topological dataset. The advantages of feature selection are more evident on topological dataset, because it has more features. This time algorithm selected the following features: all four topology-based features, is_year, is_faculty, is_dorm, from_major and to_major. On figure 5 it is clearly shown that topology-based features are far more important than other node-based features. This is also consistent with correlation matrix, since topology-based features have the highest correlations with labels. They are so informative due to phenomenon called triadic closure. The triadic closure states that in social networks connections tend to form between people who share common friends, which is precisely what these topology-based features are describing. Among the node-based features is_year, is_faculty and is_dorm were selected, all having relatively high correlation with label. Particularly high correlation has is_year, which is expected, as college students often form friendships with their classmates. Because of this from_major and to_major are also relevant. Feature is_faculty exploits the fact that students' and professors' social circles are rarely overlapping.

Embedding dataset. Feature selection on embedding dataset was especially hard due to artificial features from node2vec. Because hyperparameters of node2vec were carefully tuned, we assumed that node embeddings are optimal, hence we filtered
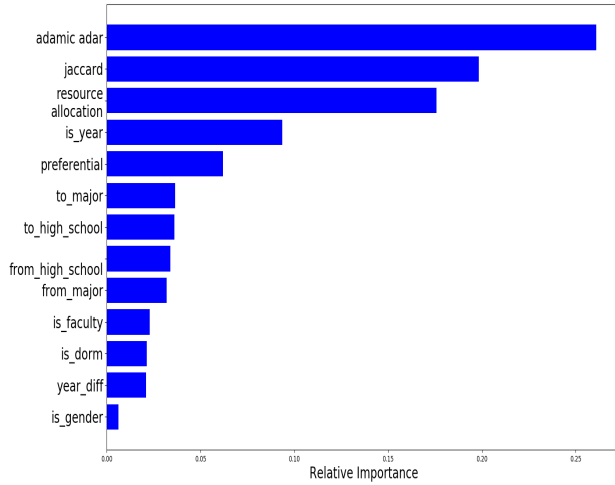
Fig. 5. Feature importance of topological dataset according to random forest classifier

only node-based features. (25). Therefore, we are using only a few crucial ones: is_year and is_dorm. We did not select is_faculty, although it is more important than is_dorm if considered on its own. We decided so, because is_faculty has high correlation with is_year and correlated features usually have negative impact on performance of a model.

Results

Evaluation of our datasets was conducted using an ensemble of classification models. We used simpler models like logistic regression and random forest, as well as more complex ones – support vector machines (SVM) and neural networks (NN). The latter are capable of modeling more complex non-linear functions, whilst logistic regression can model only the linear ones. Link prediction task was tested on all three datasets, on test and unseen data, and all aforementioned models. Performance of the models was evaluated using Area Under the Receiver Operating Characteristics (AUROC), which is one of the most common evaluation metrics for link prediction.

Dataset	Logistic regression	Random forest	SVM	NN
Baseline	0.9401	0.9227	**0.9628**	**0.9618**
Topological	0.9570	0.9173	**0.9639**	**0.9623**
Embedding	**0.9365**	0.9145	**0.9414**	**0.9389**

Table 2. AUROC values for logistic regression, random forest, support vector machine (SVM) and neural network (NN) on test data

Dataset	Logistic regression	Random forest	SVM	NN
Baseline	0.9263	0.9031	**0.9570**	**0.9560**
Topological	0.9478	0.8901	**0.9563**	**0.9538**
Embedding	**0.9229**	0.9047	**0.9217**	**0.9218**

Table 3. AUROC values for logistic regression, random forest, support vector machine (SVM) and neural network (NN) on unseen data

Table 2 contains AUROC scores for all combinations of

the datasets and models on the test data. Similarly, table 3 states the same values, but on unseen data. These tables reveal that support vector machine (SVM) and neural network (NN) are the best models for the link prediction task. Their performance is almost exactly the same, although they are based on completely different concepts. This is indicating that all relevant information from datasets is used for prediction. Only a little worse did logistic regression, which is very surprising, since it is much simpler than SVM and NN. Even more unexpected is that it outperformed random forest, which is non-linear model. This is a consequence of linear separability of the data, but more about this will be written in discussion section. All models appear to be stable, since there is only a slight decrease in performance, when applied to unseen data. Difference is negligible for baseline and topological dataset, whereas noteworthy on embedding dataset.

The models were able to extract more useful information from topological dataset than baseline and embedding ones. Baseline dataset has only a bit worse results, showing that additional node-based features have minimal influence on performance. Difference is visible for logistic regression, whereas SVM and NN have the same score on both datasets. Shockingly, embedding dataset gets the worst results. However, this might be the consequence of the chosen evaluation metric. Embedding dataset gets worse AUROC scores than the other two datasets, but better $F1$ scores. For example logistic regression on embedding dataset gets 0.87 $F1$ score, while baseline and topological datasets get only 0.76 and 0.79.

Model analysis

Here we used previously obtained data and results to optimize our models. Prior to that, data was standardized to have variance 1.0 and mean 0.0. With this approach, we have done model analysis to interpret best combinations of hyperparameters, which are useful to understand and discover patterns in data.

Logistic regression. Using grid search cross-validation on logistic regression we saw that different approaches are using different configurations. In the case of the baseline approach features equally impact the decision process, which is reflected in features' coefficients. For this dataset we used ridge regularization (L_2) with default regularization strength of 1.

In the case of topological dataset we notice that regularization here is a significant part of the process. We used lasso regularization (L_1) regularization with immense regularization strength of 1000. In this case regularization is crucial for preventing overfitting. Having in mind that our features are measurements which are not calculated in a fixed interval we are benefiting from the L_1 property of data sparsity. Model coefficients are imbalanced, where major study features (e.g. from_major and to_major) are given low coefficient values, which shows that most of the information is contained in the rest of the features.

In embedding dataset we noticed that addition of node-based data does not have any benefits. Having in mind that vectors form a feature set, we can argue that correlations inside these vectors possess structural information which is used in model learning. In this case $L1$ regularization with strength 150 is used. Lasso regularization is improving model performance on unseen networks. This is achieved with gener-

alization of the obtained knowledge from social network onto new unseen networks. As expected, coefficients show that all attributes of embedding vectors are equally important.

Random forest. Random forest did not perform well on our datasets. We can justify that conclusion by the fact that this approach lacks mechanism for regularization. Higher number of dimensions in respect to number of samples (unbalanced training and unseen data) is causing our decision tree models to overfit. Grid search in this case did not yield specific results, as well as tuning of parameters failed to find feature dependent information. This behaviour is shown in our model comparison where it is expected to experience better benchmarks on different linear models such as logistic regression and SVM. We notice that unseen networks' AUROC scores are the lowest over all datasets, therefore we can conclude that random forest model did not respond well to our problem.

Support vector machine (SVM). For support vector machine (SVM) only kernels were carefully tuned. Best fit for each dataset was chosen using grid search. Grid search consisted of linear, polynomial and Gaussian kernel, so the model could work with arbitrary dimensional data. It turned out that for baseline and topological datasets linear kernel was the best option, while embedding dataset required Gaussian kernel. This is so, because baseline and topological datasets are linearly separable as stated in discussion section, in contrast to embedding dataset, which uses node2vec that generates non-linear node embeddings.

Neural network (NN). Choosing the right hyperparameters for neural networks (NN) was very complicated and tedious task, since neural networks have a lot of different parameters. Nevertheless, correctly setting them can yield much better performance in comparison to other models. For some of the parameters like loss and optimization functions default settings were selected. Because learning-based link prediction is a binary classification task, binary cross-entropy loss function and Adam optimizer were utilized. Hidden and output activation functions were selected using random experimentation. The best results yielded ReLU as hidden activation function and sigmoid as output activation function. Lastly, architecture of neural network had to be defined, which was done using grid search. We tried a great variety of different depths and numbers of nodes per layer, but in the end architectures with only two hidden layer and small number of nodes were the best performing. Deeper architectures did not work well on topological and baseline datasets, because data is linearly separable, as explained in discussion. However, it is interesting that even on embedding dataset, which is non-linear, architectures with smaller number of layers worked better. This indicates there must be some kind of linearity even in the embedding dataset.

Discussion

Embedding dataset yielded worse results than topological and baseline ones because they are linearly separable. It is much easier to train models on linearly separable data than complex ones with a lot of non-linearity, such as node2vec. However, the embedding dataset is more stable according to $F1$ score as mentioned in the results.

Random forest is the worst on baseline and topological data, because it is harder for it to adjust to linear data. Other
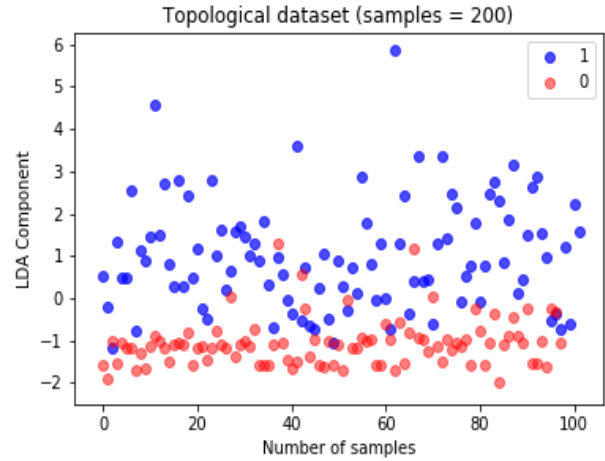

Fig. 6. LDA on topological dataset

models are capable of that, while logistic regression is the linear model by default. To contribute to that, SVM works best while using the linear kernel. Neural network works better if it has only two hidden layers, which indicates that the model needs to content some sort of linearity. To prove this we can use linear discriminant analysis (**LDA**) to visualize our data. If any linearity coexists in the dataset, it should be visible in reduced dimension space. With this representation we can truly see that the data is more linear in topological data (figure 6). Comparing to embedding data, we can see that we have higher non linearity (figure 7). Having that in mind, slightly expressed non-linearity can be suppressed with regularization and result in a trade-off, resulting in increased stability of the prediction model.
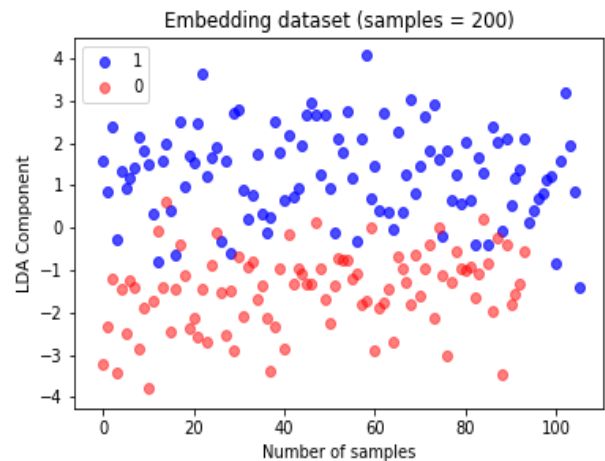

Fig. 7. LDA on embedding dataset

Conclusion

In the presented paper we can conclude that results are unexpectedly good for link prediction tasks of this nature. Having in mind that we are generating predictions learned from separated social networks we can say that our models succeeded

in their task. It is visible that models successfully generalized unseen data, based only on the 50% bigger training set.

For optimization of AUROC score baseline and topological approaches are the best. It turns out that simplicity has benefits in terms of high classification scores. In these two cases node-based features did not really effect performance, except for logistic regression, where binary data was utilized in a manner where linearity was enlarged. Although SVM and NN got better results, we recommend logistic regression in combination with topological approach because the model is easier to train and interpret. When very high AUROC scores are important (e.g. link prediction on medical data), we suggest SVM with linear kernel and baseline approach. It gets almost the same results on unseen data, even though it is simpler model than NN.

We have shown that collecting data from multiple social networks yields promising datasets, which can be used for modelling of various predictors in similar social structures. Besides that, in this paper we have shown that use of regularization can be a solution in the case of social networks, when lack of the training data is present. Using this approach we can obtain data insights globally.

References

1. P Wang, B Xu, Y Wu, X Zhou, Link prediction in social networks: the state-of-the-art. *Sci. China Inf. Sci.* **58** (2014).
2. J Scott, Social network analysis. *Sociology* **22** (1988).
3. GKELRR Pow, J., Understanding complex interactions using social network analysis. *J. Clin. Nurs.* **1**, 2779 (2012).
4. MA Hasan, V Chaoji, S Salem, M Zaki, Link prediction using supervised learning in *In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security.* (2006).
5. L Breiman, Machine learning, volume 45, number 1 - springerlink. *Mach. Learn.* **45**, 5–32 (2001).
6. C Cortes, V Vapnik, Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).
7. S Scellato, A Noulas, C Mascolo, Exploiting place features in link prediction on location-based social networks in *KDD.* (ACM), pp. 1046–1054 (2011).
8. MEJ Newman, Clustering and preferential attachment in growing networks. *Phys. review. E, Stat. nonlinear, soft matter physics* **64 2 Pt 2**, 025102 (2001).
9. G Salton, MJ McGill, *Introduction to Modern Information Retrieval.* (McGraw-Hill, Inc., USA), (1986).
10. L Adamic, E Adar, Friends and neighbors on the web. *Soc. Networks* **25**, 211–230 (2003).
11. A Barabási, et al., Evolution of the social network of scientific collaborations. *Phys. A: Stat. Mech. its Appl.* **311**, 590–614 (2002).
12. L Lü, CH Jin, T Zhou, Similarity index based on local paths for link prediction of complex networks. *Phys. review. E, Stat. nonlinear, soft matter physics* **80**, 046122 (2009).
13. L Katz, A new status index derived from sociometric analysis. *Psychometrika* **18**, 39–43 (1953).
14. G Jeh, J Widom, Simrank: A measure of structural-context similarity in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02. (Association for Computing Machinery, New York, NY, USA), p. 538–543 (2002).
15. H Liu, Z Hu, H Haddadi, H Tian, Hidden link prediction based on node centrality and weak ties. *EPL (Europhysics Lett.* **101**, 18004 (2013).
16. T Mikolov, K Chen, G Corrado, J Dean, Efficient estimation of word representations in vector space. *Proc. Work. at ICLR* **2013** (2013).
17. B Perozzi, R Al-Rfou, S Skiena, Deepwalk: Online learning of social representations in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14. (Association for Computing Machinery, New York, NY, USA), p. 701–710 (2014).
18. A Grover, J Leskovec, Node2vec: Scalable feature learning for networks in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16. (Association for Computing Machinery, New York, NY, USA), p. 855–864 (2016).
19. Z Zhang, et al., Arbitrary-order proximity preserved network embedding in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, KDD '18. (Association for Computing Machinery, New York, NY, USA), p. 2778–2786 (2018).
20. AL Traud, PJ Mucha, MA Porter, Social structure of Facebook networks. *Phys. A* **391**, 4165–4180 (2012).
21. AL Traud, ED Kelsic, PJ Mucha, MA Porter, Comparing community structure to characteristics in online collegiate social networks. *SIAM Rev.* **53**, 526–543 (2011).
22. RA Rossi, NK Ahmed, The network data repository with interactive graph analytics and visualization in *AAAI.* (2015).
23. T Zhou, L Lü, YC Zhang, Predicting missing links via local information. *The Eur. Phys. J. B - Condens. Matter Complex Syst.* **71**, 623–630 (2009).
24. A Jovic, K Brkic, N Bogunovic, A review of feature selection methods with applications. *2015 38th Int. Conv. on Inf. Commun. Technol. Electron. Microelectron. (MIPRO)*, 1200–1205 (2015).
25. HZ L. Liao, X. He, TS Chua, Attributed social network embedding. *TKDE* (2018).