Twitch Gamers: a Dataset for Evaluating Proximity Preserving and Structural Role-based Node Embeddings

Benedek Rozemberczki
The University of Edinburgh
Edinburgh, United Kingdom
benedek.rozemberczki@ed.ac.uk

Rik Sarkar
The University of Edinburgh
Edinburgh, United Kingdom
rsarkar@inf.ed.ac.uk

ABSTRACT

Proximity preserving and structural role-based node embeddings have become a prime workhorse of applied graph mining. Novel node embedding techniques are often tested on a restricted set of benchmark datasets. In this paper, we propose a new diverse social network dataset called *Twitch Gamers* with multiple potential target attributes. Our analysis of the social network and node classification experiments illustrate that *Twitch Gamers* is suitable for assessing the predictive performance of novel proximity preserving and structural role-based node embedding algorithms.

1 INTRODUCTION

The prediction of unknown node attributes using vertex features is a central problem in both theoretical and applied graph mining research. One way to create high quality node features is to embed the vertices in an Euclidean space. Node embedding algorithms are frequently used as an upstream unsupervised feature extraction method to distill useful features for downstream supervised models. Their success is mainly due to the favorable algorithmic qualities they have such as runtime and memory efficiency. In addition to efficiency, the extracted node representations are known to be robust to hyperparameter changes [12, 13, 16] and the learned features are reusable when new downstream machine learning tasks come up [1, 7]. Node embedding techniques are typically evaluated on a limited number of public benchmark datasets [7, 8, 12–14, 21], which are not compatible with newly proposed attribute based algorithms [15, 16, 20]. This highlights the need for new benchmark datasets which are rich in attributes.

Present work. In order to foster node embedding research we publicly release *Twitch Gamers*: a medium sized undirected social network of online streamers with multiple interesting vertex attributes. Using *Twitch Gamers*, the predictive performance of a node embedding algorithm can be tested on multiple new challenging node classification and vertex level regression problems. Potential machine learning tasks include the identification of dead accounts, selection of users that stream explicit content and broadcaster language prediction. Our work creates opportunity for the assessment of numerous existing node representation learning techniques and newly developed vertex embedding procedures.

Main contributions. The most important contributions of our paper can be summarized as follows:

(1) We release *Twitch Gamers*: a new social network dataset which we specifically collected for benchmarking the vertex classification performance of proximity preserving and structural role-based node embedding techniques.
(2) We carry out a descriptive analysis of the social network and underlying generic vertex features and argue that it is suitable for testing novel node embedding methods.
(3) We evaluate the performance of standard node embedding algorithms under various train/test split regimes.

The rest of our work has the following structure. We overview the related work about node embedding procedures in Section 2. We discuss in Section 3 the data collection and the dataset itself. We perform descriptive analysis of the social network and the generic vertex attributes in Section 4. In Section 5 we showcase the predictive performance of various well known node embedding techniques on the *Twitch Gamers* dataset. The paper concludes with Section 6 where we discuss potential future work.

2 RELATED WORK

Given a graph $G = (V, E)$ node embedding techniques learn a function $f : V \longrightarrow \mathbb{R}^d$ which maps the nodes $v \in V$ into a d dimensional Euclidean space. When generic vertex features are not available for node classification, proximity preserving and structural role-based node embedding techniques are suitable for distilling high quality reusable feature sets [1]. We will utilize linear runtime node embedding algorithms to showcase that *Twitch Games* is suitable for multi-aspect testing of feature extraction.

Proximity preserving node embedding algorithms [6, 10, 12, 13, 16] learn this embedding by preserving a certain notion of proximity in the embedding space such as pairwise truncated random walk transition probabilities. This way nodes that are close to each other in the graph are also close in the embedding space. Structural role-based node embedding techniques on the other hand preserve structural similarity in the embedding space. Nodes which have similar structural properties such as centrality and transitivity are close to each other in the embedding space [5, 7, 8].

3 THE TWITCH GAMERS DATASET

Twitch is a streaming service where users can broadcast live streams of playing computer games. As users can follow each other there is an underlying social network which can be accessed through the public API. In 2018 April we crawled the largest connected component of this social network with snowball sampling starting from the user called *Lowko*. The released *Twitch Gamers* dataset is a clean subset of the original social network. We filtered out nodes and edges based on the following principled steps:

(1) **No missing attributes.** We only kept nodes that have all of the vertex attributes present.

(2) **Mutual relationships.** We discarded relationships which are asymmetric and only included mutual edges in the released dataset.
(3) **Member of the largest component.** We only considered nodes which are part of the largest connected component.

The result of this three step data cleaning process is an undirected, single component social network with approximately 168 thousand nodes and 6.79 million edges. Vertices in this restricted subsample do not have any missing node attributes. We summarized the name, meaning, and type of available generic node attributes in Table 1.

Name	Meaning	Type
Identifier	Numeric vertex identifier.	Index
Dead Account	Inactive user account.	Categorical
Broadcaster Language	Languages used for broadcasting.	Categorical
Affiliate Status	Affiliate status of the user.	Categorical
Explicit Content	Explicit content on the channel.	Categorical
Creation Date	Joining date of the user.	Date
Last Update	Last stream of the user.	Date
View Count	Number of views on the channel.	Count
Account Lifetime	Days between first and last stream.	Count

Table 1: The name, meaning and type of vertex attributes in the *Twitch Gamers* dataset.

Categorical attributes such as *Dead Account, Affiliate Status*, and *Explicit Content* can be used as targets for binary classification, while *Broadcaster Language* can be used for multi-class node classification with more than 20 categories. The vertex attributes *View Count* and *Account Lifetime* can serves as target for count data regression problems at the node level. Various other supervised and unsupervised machine learning tasks can be performed on the dataset such as link prediction and community detection with ground truth labels. The *Twitch Gamers* dataset (edge list and generic vertex attributes) is publicly available at https://github.com/benedekrozemberczki/datasets.

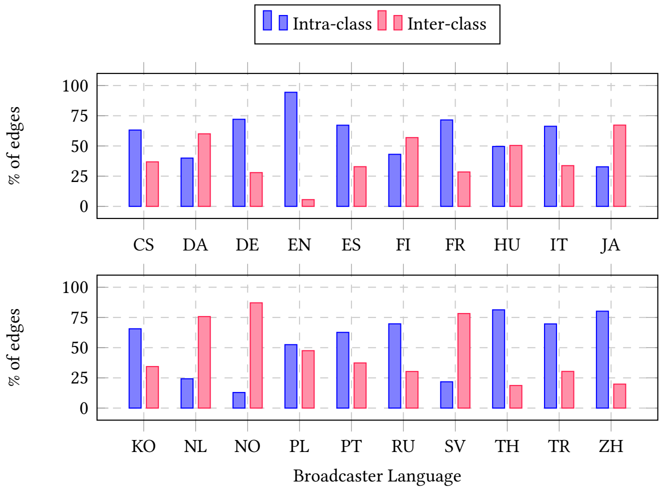

Figure 1: The percentage of intra and inter-class edges conditional on the *Broadcaster Language* attribute.

4 DESCRIPTIVE ANALYSIS

Our descriptive analysis of *Twitch Gamers* focuses on the interaction of graph topology and attributes. Specifically, we investigate which potential target attributes can be predicted well with neighbourhood-preserving and structural role-based techniques.

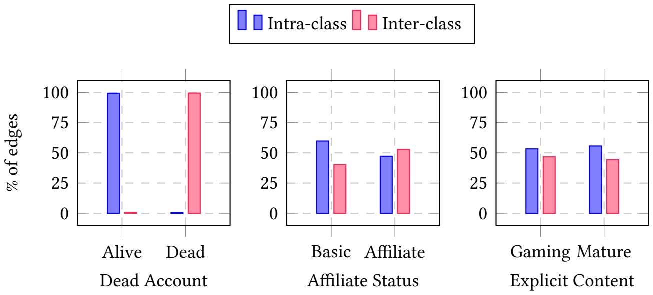

Figure 2: The percentage of intra and inter-class edges conditional on the *Dead Account, Affiliate Status*, and *Explicit Content* attributes.

We plotted the ratio of inter and intra-class edges conditional on the categorical attributes on Figures 1 and 2. These results show that users who broadcast in more commonly spoken language (English, German, French) are more likely to have connections with users who broadcast in the same language. This postulates that proximity preserving node embedding techniques will extract expressive features that can predict *Broadcaster Language* precisely. We also see that Twitch users who churned from the platform are well embedded in the social network and do not form communities. When it comes to the *Affiliate Status* and *Explicit Content* attributes we cannot highlight particular insights about the related linking behaviour of vertices.

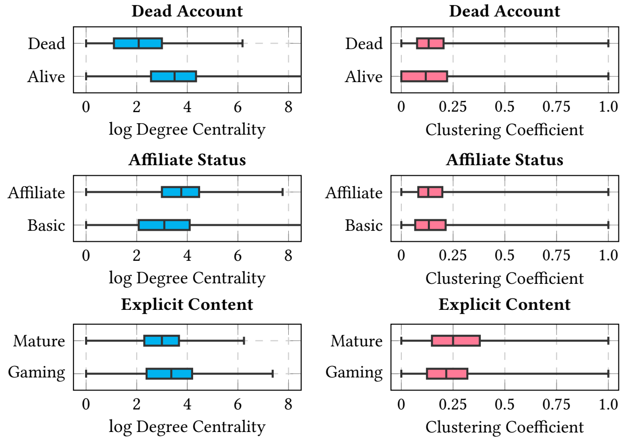

Figure 3: The box plots of degree centrality and clustering coefficient conditional on the *Dead Account, Affiliate Status*, and *Explicit Content* attributes.

We used boxplots to visualize the distribution of the log transformed degree and clustering coefficient conditional on the categorical vertex attributes. We plotted these boxplots of the structural features on Figures 4 and 3. Based on these plots we can deduce that users who broadcast in more commonly spoken languages are well connected. At the same time their friends are less likely to be connected – this potentially hints at their hub-like role. The results obtained for the other attributes are also intuitive: (i) users who churned from the platform are less central in the social network; (ii) broadcasters who use explicit language are less popular; (iii) those who obtain affiliate status are generally well connected in the social network. These findings hint that all of the categorical features can be embedded with the use of structural role-based node embedding techniques.

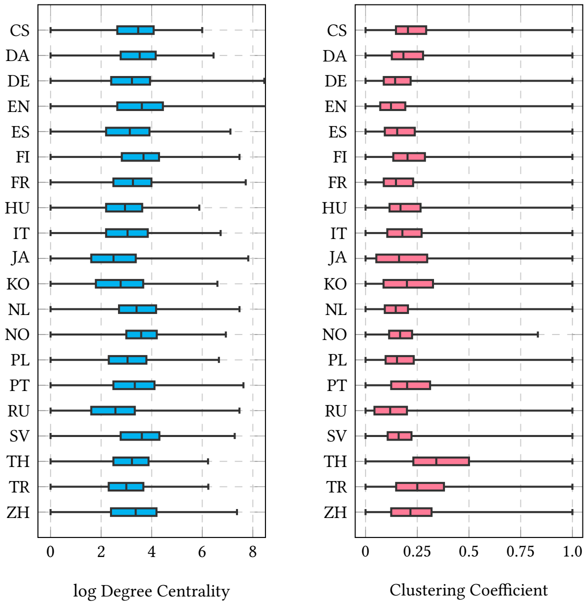

Figure 4: The box plots of degree centrality and clustering coefficient conditional on the *Broadcaster Language* attribute.

5 EXPERIMENTAL EVALUATION

We use *Twitch Gamers* to evaluate the predictive value of features extracted with popular node embedding algorithms. The target attributes of node classification were the *Explicit Content*, *Broadcaster Language*, *Dead Account* and *Affiliate Status* variables. In our experiments we used the open-source *Karate Club* [17] library with the default hyperparameter settings of the node embedding procedures. Specifically, we tested the performance of the following proximity preserving node embeddings:

(1) **Diff2Vec** [18, 19] factorizes a pointwise mutual information (henceforth PMI) matrix derived from a diffusion process.
(2) **DeepWalk** [12] decomposes the PMI matrix of summed normalized adjacency matrix powers with implicit factorization.
(3) **Walklets** [13] factorizes the PMI matrix of normalized adjacency matrix powers to obtain multi-scale node embeddings.
(4) **RandNE** [22] smooths an orthogonal node embedding matrix with powers of the adjacency matrix.

We evaluated the value of features extracted with these structural role-based node embedding algorithms:

(1) **Role2Vec** [2] decomposes the PMI matrix node – tree feature co-occurrences with an implicit factorization technique.
(2) **ASNE** [9] factorizes a target matrix obtained by concatenating the adjacency matrix and a structural feature matrix which includes one-hot encodings of the log degree and clustering coefficient.
(3) **MUSAE** [15] learns multi-scale structural role-based node embeddings from matrices obtained by multiplying the structural feature matrix with adjacency matrix powers.
(4) **FEATHER** [20] distills node embeddings from graph characteristic functions of the log transformed degree and clustering coefficient.

We used the *scikit-learn* [3, 11] implementation of logistic regression with the default hyperparameter settings to predict the node labels using the node embeddings as input features. It has to be noted that these default settings involve the use of weight regularization, because of this each node embedding dimension was normalized. The classifiers were trained with various highly skewed train/test data split ratios by utilizing less than 1% of training data. We plotted mean macro-averaged AUC scores on the test set calculated from 10 random seed train/test splits on Figures 5 and 6.

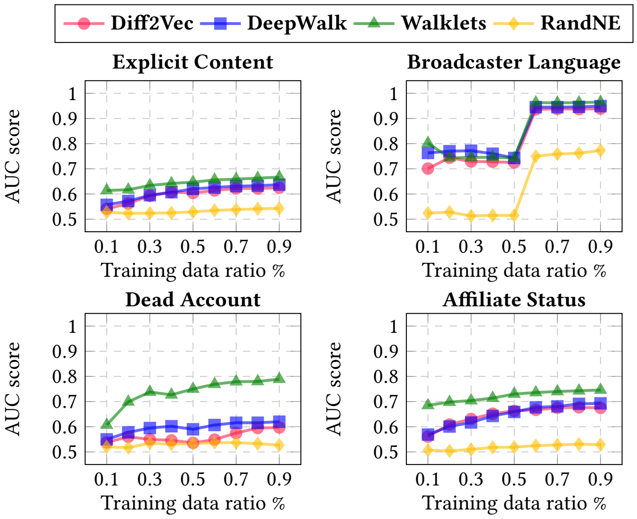

Figure 5: Predictive performance of proximity preserving node embedding techniques on classification tasks measured by area under the curve scores on the test set as a function of training set ratio.

The most important finding based on our results is that the target attributes in *Twitch Gamers* are suitable for testing the predictive power of features extracted with both proximity preserving and structural role-based node embeddings. These results showcase that certain node embedding techniques have a considerable advantage on the downstream tasks. We also see evidence that proximity preserving algorithms extract features which are more useful for predicting *Broadcaster Language*. This was expected based on our empirical analysis as it is an attribute which most probably strongly influences linking behaviour. Another similar intuitive finding is that structural role-based embedding techniques[2] create more expressive features for predicting the *Dead Account* target variable. This is not surprising, our descriptive analysis had shown that users who churned from the platform have idiosyncratic structural attributes. Our results also verify the known fact that multi-scale proximity preserving node embeddings, such as *Walklets* [13] and *GraRep* [4], outperform techniques like *DeepWalk* [12] that pool information form low and higher order proximities.

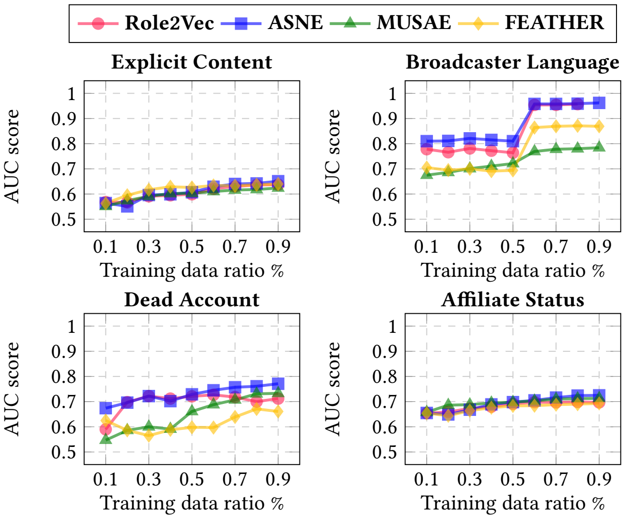

Figure 6: Predictive performance of structural role preserving node embedding techniques on classification tasks measured by area under the curve scores on the test set as a function of training set ratio.

6 CONCLUSIONS

We introduced *Twitch Gamers* a medium sized social network dataset with a rich set of potential target attributes. Our descriptive analysis of the dataset had demonstrated that both proximity preserving and structural role-based node embeddings can potentially distill high quality features for node classification. We verified this precognition by a series of experiments. Our findings show that *Twitch Gamers* can serve as an important benchmark to assess novel node embedding techniques. We are particularly excited that the prediction of certain vertex attributes turned out to be challenging machine learning task.

ACKNOWLEDGEMENTS

Benedek Rozemberczki was supported by the Centre for Doctoral Training in Data Science, funded by EPSRC (grant EP/L016427/1).

REFERENCES

[1] N. Ahmed, R. A. Rossi, J. Lee, T. Willke, R. Zhou, X. Kong, and H. Eldardiry. 2020. Role-based Graph Embeddings. *IEEE Transactions on Knowledge and Data Engineering* (2020), 1–1.

[2] Nesreen K Ahmed, Ryan Rossi, John Boaz Lee, Xiangnan Kong, Theodore L Willke, Rong Zhou, and Hoda Eldardiry. 2018. Learning Role-based Graph Embeddings. *Proceedings of the 26th IJCAI Conference on Artificial Intelligence - Statistical Relational AI Workshop* (2018).

[3] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API Design for Machine Learning Software: Experiences from the Scikit-Learn Project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. 108–122.

[4] Shaosheng Cao, Wei Lu, and Qiongkai Xu. 2015. GraRep: Learning Graph Representations with Global Structural Information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 891–900.

[5] Claire Donnat, Marinka Zitnik, David Hallac, and Jure Leskovec. 2018. Learning Structural Node Embeddings via Diffusion Wavelets. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1320–1329.

[6] Aditya Grover and Jure Leskovec. 2016. Node2Vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 855–864.

[7] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. 2012. RolX: Structural Role Extraction and Mining in Large Graphs. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1231–1239.

[8] Keith Henderson, Brian Gallagher, Lei Li, Leman Akoglu, Tina Eliassi-Rad, Hanghang Tong, and Christos Faloutsos. 2011. It's Who You Know: Graph Mining Using Recursive Structural Features. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*. ACM, 663–671.

[9] Lizi Liao, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2018. Attributed Social Network Embedding. *IEEE Transactions on Knowledge and Data Engineering* 30, 12 (2018), 2257–2270.

[10] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. 2016. Asymmetric Transitivity Preserving Graph Embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1105–1114.

[11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research* 12, Oct (2011), 2825–2830.

[12] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM.

[13] Bryan Perozzi, Vivek Kulkarni, Haochen Chen, and Steven Skiena. 2017. Don't Walk, Skip!: Online Learning of Multi-scale Network Embeddings. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*. ACM, 258–265.

[14] Jiezhong Qiu, Yuxiao Dong, Hao Ma, Jian Li, Kuansan Wang, and Jie Tang. 2018. Network Embedding as Matrix Factorization: Unifying DeepWalk, LINE, PTE, and Node2Vec. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*. ACM, 459–467.

[15] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. 2019. Multi-Scale Attributed Node Embedding. *arXiv preprint arXiv:1909.13021* (2019).

[16] Benedek Rozemberczki, Ryan Davies, Rik Sarkar, and Charles Sutton. 2019. GEMSEC: Graph Embedding with Self Clustering. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2019*. ACM, 65–72.

[17] Benedek Rozemberczki, Oliver Kiss, and Rik Sarkar. 2020. Karate Club: An API Oriented Open-source Python Framework for Unsupervised Learning on Graphs. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*. ACM.

[18] Benedek Rozemberczki, Oliver Kiss, and Rik Sarkar. 2020. Little Ball of Fur: A Python Library for Graph Sampling. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*. ACM, 3133–3140.

[19] Benedek Rozemberczki and Rik Sarkar. 2018. Fast Sequence-Based Embedding with Diffusion Graphs. In *International Workshop on Complex Networks*. Springer, 99–107.

[20] Benedek Rozemberczki and Rik Sarkar. 2020. Characteristic Functions on Graphs: Birds of a Feather, from Statistical Descriptors to Parametric Models. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*. ACM.

[21] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-Scale Information Network Embedding. In *Proceedings of the 24th International Conference on World Wide Web*. 1067–1077.

[22] Ziwei Zhang, Peng Cui, Haoyang Li, Xiao Wang, and Wenwu Zhu. 2018. Billion-scale network embedding with iterative random projection. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 787–796.