VENUERANK: IDENTIFYING VENUES THAT CONTRIBUTE TO ARTIST POPULARITY

Emmanouil Krasanakis[1] **Emmanouil Schinas**[1,2]
Symeon Papadopoulos[1] **Yiannis Kompatsiaris**[1] **Pericles Mitkas**[2]
[1]CERTH-ITI, Thessaloniki, Greece - {maniospas,manosetro,papadop,ikom}@iti.gr
[2]AUTH, Thessaloniki, Greece - manosetro@issel.ee.auth.gr,mitkas@auth.gr

ABSTRACT

An important problem in the live music industry is finding venues that help expose artists to wider audiences. However, it is often difficult to obtain live music audience data to tackle this task. In this work, we investigate whether important venues can instead be inferred through social media data. Our approach consists of employing bipartite graph ranking algorithms to help discover important venues in artist-venue graphs mined from Facebook. We use both well-established algorithms, such as BiRank, and a modification of their common iterative scheme that avoids the impact of possibly erroneous heuristics to the ranking, which we call VenueRank. Resulting venue ranks are compared to those obtained from feature extraction for predicting the most listened artists and large listener increments in Spotify. This comparison yields high correlation between venue importance for listener prediction and bipartite graph ranking algorithms, with VenueRank found more robust against overfitting.

1. INTRODUCTION

In the music industry, artists aim to present themselves to wide audiences. Therefore, it is important for them to gain as much exposure as possible from their live performances. In turn, this exposure can influence their popularity, as expressed by the size of their audience.

In this paper we try to identify which performances offer higher exposure. Factors such as timing and other recent events can influence this. Listener geolocation has also been found to contribute to artist popularity prediction [3, 21]. Consequently, it is reasonable to hypothesize that performing in certain venues could contribute more to artist popularity. Having access to a ranking of venues based on expected exposure could be valuable for artists and their agents; when confronted with different options regarding their future performances, they could consider these rankings as an important decision criterion.

To rank venue exposure, one could try to predict it using machine learning algorithms. Unfortunately, there are difficulties in quantifying the notion of exposure, not least of which is that real-life data may misrepresent audience size and reactions. For example, participating in a large event held in a well-known venue with many other artists may contribute less to gaining popularity compared to an artist-focused event. A viable alternative to measuring venue exposure, which we also adopt in this work, is to instead estimate whether venues contribute to artist popularity (e.g. the number of listeners in music services) from a machine learning perspective. To do so, we can employ feature extraction methods to identify the most important venues that help determine and increase artist popularity.

However, even this formulation depends on obtaining live music audience data required for supervised training. Such data are not necessarily easy to obtain, as they are typically considered confidential. Therefore, in this work we attempt inferring important venues through unsupervised training, which does not require such data.

In particular, given a graph representation, where artists are linked with venues they have performed in, we use graph ranking algorithms to rank venues. To validate whether this approach ranks venues based on offered exposure, we compare the produced ranks with venue importances obtained through feature extraction for predicting popular artists and artist popularity increments. We find that ranking methods can be more informative than raw social media measures in predicting important venues.

2. BIPARTITE GRAPH RANKING

2.1 Motivation for Graph Ranking

We can organize data pertaining to artists $\mathcal{A}$ and venues $\mathcal{V}$ where they have performed as bipartite graphs, i.e. graphs in which vertices form two disjoint sets linking only to each other. To analyze the importance of venues based on the structure of such artist-venue graphs, we employ ranking algorithms, which are used to determine the relative importance of nodes given a graph's structure [17]. These algorithms often operate under the premise that nodes linked to a higher number of important nodes are also more important.

Formulations such as HITS [14] further refine this concept by recognizing that there can be two types of important nodes; authorities that provide important information and hubs that point to a lot of information sources. A node's authority is then derived from its predecessors' hub scores and its hub score is derived from its successors' authority scores, forming an iterative process.

The distinction between authorities and hubs is of great interest when applied on bipartite graphs, especially if the links between disjoint set elements represent the same type of relation. In this case, we can formulate that one of those sets contains only authorities and the other only hubs.

For example, in our artist-venue graph setting, where venues always represent locations where artists have performed, artists can be considered as authorities and venues as hubs from which popularity-related authority stems. Intuitively, this means that artists are considered more important if they have performed in more important venues, whereas venues are considered more important if more important artists have performed there.

2.2 Ranking based on Prior Ranks

Formally, bipartite graph ranking algorithms attempt to rank nodes in a graph defined by a (weighted) adjacency matrix $W : \mathcal{V} \times \mathcal{A}$ between the disjoint groups $\mathcal{A}, \mathcal{V}$ based on heuristically estimated prior ranks.

Prior ranks [1] are supported by most graph ranking algorithms and are used to introduce ranking bias that is driven by information unrelated to graph structure. For example, in web searches [17] prior ranks place more weight on the pages more similar to the search query. In bipartite graphs, prior ranks often represent an informed belief about ranks and help attract the solution towards convergence. However their usage can reduce the robustness of extracted structural characteristics (see Subsection 2.3).

As demonstrated by He et al. [12], previous bipartite graph ranking algorithms follow similar formulations. In particular, if S and S' are normalizations of W and its transposition W^T respectively, a_0, v_0 are prior ranks that initially estimate node ranks for the two bipartite graph groups and r_a, r_v are prior rank elimination parameters, approaches follow a common recursive rule for calculating bipartite graph group ranks as recursions $n \to \infty$:

$$a_{n+1} = (1 - r_a)a_0 + r_a S' v_n \quad (1a)$$

$$v_{n+1} = (1 - r_v)v_0 + r_b S a_n \quad (1b)$$

In practice, this iterative process stops when rank differences converge to a stable set of values. In this work, we empirically adopt a simple stopping criterion across algorithms that stops after rank changes become small enough:

$$\|a_{n+1} - a_n\|_2^2 + \|v_{n+1} - v_n\|_2^2 < 0.1 \quad (2)$$

Differences between bipartite graph ranking algorithms lie in the way normalization is performed on the adjacency matrix and its transposition. If $D_{\mathcal{A}}, D_{\mathcal{V}}$ are two diagonal matrices containing the node degrees of the disjoint sets $\mathcal{A}, \mathcal{V}$, those algorithms perform normalization as:

$$S = D_v^{-p_v} W D_a^{-p_a} \quad (3a)$$

$$S' = D_a^{-p_v} W^T D_v^{-p_a} \quad (3b)$$

[1] Prior ranks have also been referred to as 'query vectors' [12].

where p_a, p_v are non-negative constants specific to each algorithm (see Table 1). These constants determine whether degree normalization should be performed row-wise or column-wise or whether the normalization should produce a stochastic matrix.

Algorithm	p_a	p_v
HITS [14]	0	0
Co-HITS [6]	1	0
BGRM [19]	1	1
BGER [1]	0	1
BiRank [12]	$\frac{1}{2}$	$\frac{1}{2}$

Table 1: Different parameters between bipartite graph ranking algorithms.

The advantage of the iterative scheme demonstrated in Eqn (1) over more general ranking schemes, which do not take the bipartite nature of the graph into account, is that the former converges fast to unique stationary solutions, as demonstrated below.

a) If $r_a, r_v < 1$ then substituting Eqn (1) into itself as $n \to \infty$ yields:

$$a_\infty = (I - r_a r_v S'S)^{-1}[r_a(1 - r_v)S'v_0 + (1 - r_a)a_0]$$
$$v_\infty = (I - r_a r_v SS')^{-1}[r_v(1 - r_a)Sa_0 + (1 - r_v)v_0]$$

Although this solution can also help analytically derive node ranks a_∞, v_∞, doing so can be computationally intensive, since it requires matrix inversion. For that reason, all previous approaches adopt the iterative scheme, which is computationally efficient, especially when W is sparse.

b) If $r_a = r_v = 1$ then:

$$\begin{matrix} a_\infty = S'v_\infty \\ v_\infty = Sa_\infty \end{matrix} \Rightarrow \begin{matrix} (S'S - I)a_\infty = 0 \\ (SS' - I)v_\infty = 0 \end{matrix}$$

Therefore, if $S'S, SS'$ are stochastic matrices, i.e. if $p_a + p_v = 1$ in Eqn (3), their largest eigenvalue is 1 and thus a_∞, v_∞ are their principal eigenvectors respectively. In this case, the iterative scheme resembles the power method for calculating the principal eigenvectors. However, due to absence of vector normalization after each step, large enough ranks may grow uncontrollably and fail to converge [16].

2.3 VenueRank

The above formulation of bipartite graph ranking algorithms relies heavily on the correctness of the prior ranks a_0, v_0 to produce accurate ranks. If the prior ranks are only partially correct (e.g. are only sparsely filled) ranking algorithms may converge to much different values. Furthermore, structure-related information could be more useful for important venue detection than prior rank heuristics. Hence, there exist cases where eliminating the effect of prior ranks is desirable [15].

In such cases, the previous bipartite graph ranking algorithms eliminate the prior ranks by selecting $r_a = r_v = 1$.

However, as discussed above, numeric convergence is not theoretically guaranteed for these parameters. For example, BiRank fails to converge for these parameters when run on data gathered in Subsection 3.1.

Therefore, we propose modifying the previous iterative process to gradually remove the dependency on initial prior ranks across iterations:

$$a_{n+1} = (1 - r_a)a_n + r_a S' v_n \quad (4a)$$

$$v_{n+1} = (1 - r_v)v_n + r_v S a_n \quad (4b)$$

Similarly to before, as $n \to \infty$ we obtain $a_\infty = S' v_\infty$ and $v_\infty = S a_\infty$ and thus a_∞, v_∞ become the principal eigenvectors of $S'S$ and SS' respectively, as long as the latter are stochastic matrices.

This iterative process differs from previous ones in that it stabilizes on these eigenvectors for any non-zero parameters r_a, r_v. As a result, we can retrieve theoretical guarantees [16] that there exist small enough r_a, r_v that make it converge. Moreover, we can see that:

$$S'S = D_a^{-p_v - p_a} W^T D_v^{-p_a - p_v} W$$

$$S'S = D_v^{-p_a - p_v} W D_a^{-p_v - p_a} W^T$$

Therefore, for constant $p_a + p_v = 1$ the eigenvectors of these two matrices remain the same and Eqn (4) converges to the same ranks regardless of the type of normalization defined by these two parameters.

In short, we have shown that, for the iterative scheme demonstrated in Eqn (4), which we will call VenueRank, it suffices to select any $p_a + p_v = 1$ and small enough r_a, r_v to converge to bipartite graph ranks where the effect of prior ranks is eliminated.

3. EXPERIMENTS

3.1 Data Collection

We collected two types of data for our experiments; data from *Facebook* about artist and venue pages and the respective number of listeners for those artists from *Spotify*. We use Facebook data to run bipartite graph ranking algorithms and Spotify data to extract the ground truth with which to evaluate these algorithms.

We started with a collection of 542 artists, for which we were granted access to their number of streams and listeners in Spotify Analytics [2] from 1 January 2015 to 3 May 2019. We also used the Facebook Graph API [3] to automatically find Facebook pages for those artists and manually removed artists with erroneously matched pages. After this step, the collection comprises 323 artists, for whom we can retrieve both the monthly number of listeners in Spotify and their Facebook page.

Next, we retrieved the events published in the discovered Facebook pages dating later than 1 January 2014, as well as the venues that hosted them. This process results in a tripartite graph with nodes representing artists, events and venues (see Figure 1).

[2] https://analytics.spotify.com/
[3] https://developers.facebook.com/docs/graph-api/

Figure 1: Artist (red) and venue (blue) pages in Facebook alongside their associated events (yellow) of the largest connected subgraph of the dataset that contains a Finnish rock band called 'The Rasmus' (green).

This graph contains a total of 105,251 events that took place in 4,051 venues across 72 timezones (see Figure 2). Using the events in that graph as indicators of the appearance of an artist in a venue, we infer a bipartite artist-venue graph, which we use for our analysis. Artists associated with a non-zero number of venues number 224 and they are associated with a total of 2,392 venues.

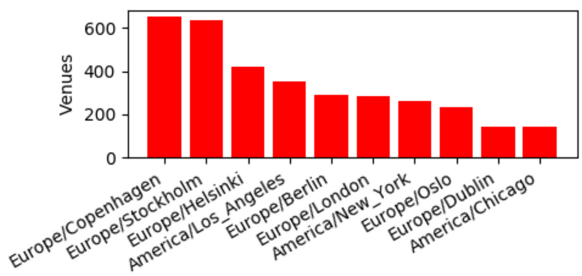


Figure 2: Timezones with more than 20 venues each.

Our dataset contains the number of Spotify listeners per month for each artist. From these listeners we procure artist popularity based on the total number of listeners for each artist, as well as the increase of the number of artist listeners for each month, which yields 8,619 datapoints across all artists. The number of total listeners spans a wide range of magnitudes. Hence, denoting the number of listeners for month m of an artist as L_m, we define:

$$popularity = \log(1 + \sum_m L_m)$$

which yields the normal-like artist distribution shown in Figure 3. We also quantify the relative increments of monthly listeners as:

$$inc_m = \min\{L_m / L_{m-1} - 1, 1\} \text{ when } L_{m-1} \neq 0$$

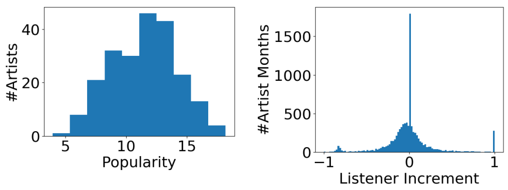


Figure 3: Artist popularities (left) and the distribution of relative listener increments (right).

3.2 Ground Truth Construction

It is difficult to directly extrapolate the exposure granted by venues using only artist $popularity$ and inc_m data. However, we can employ a feature extraction scheme to obtain venue importances reflecting their contribution to predicting these quantities. Based on the systemic property of graph ranking algorithms in Section 2.2, to rank based on the nodes' positions on the graph, we can then evaluate the quality of those algorithms by measuring whether higher ranked venues are actually more important for prediction. Effectively, this would assert that higher ranks represent higher exposure. This way, venue importances extracted from machine learning on Spotify data can be used as ground truth against which to validate ranking algorithms, which do not utilize such data.

The machine learning setting for feature extraction can be either a regression or a classification task (popular-vs-unpopular). Here, we focus on the latter, since the classification task leads to clearer separation between venues that lead to each of the two target classes. Indeed, regression tasks using only venues to predict $popularity$ and inc_m values yield high error rates, whereas the binary classifiers demonstrated below boast high predictive capabilities.

Labeling and Feature Selection

To set up the venue ranking task, we distinguish high $popularity$ and inc_m values by performing outlier detection [13] and considering outliers residing in the 20% right tail of their distributions [4] as popular and high increment ones respectively. We then use methods that perform robust feature extraction [10] based on these labels.

To do so, we consider venues as binary artist features to predict $popularity$, whereas we use exponential decay to model the decreasing influence [9] on inc_m of performing in a venue held at month m_v as $exp\big(-\frac{m-m_v}{2}\big)$ if $m_v \leq m$ and 0 otherwise. Using these feature values, the feature extraction mechanism then identifies which features (i.e. venues) contribute the most to label prediction (i.e. high artist popularity and high listener increments).

Unfortunately, outliers represent a small fraction of all datapoints and hence cause imbalance between label priors. Imbalance often affects classification validity and can distort or bias estimated feature importances. To alleviate such concerns, we employ SMOTE oversampling [2] to generate synthetic popular artist profiles, so that the number of popular artists becomes equal to the number of unpopular ones. We prefer an oversampling scheme, because the small number of collected artists prohibits an undersampling one. This process is summarized in Figure 4.

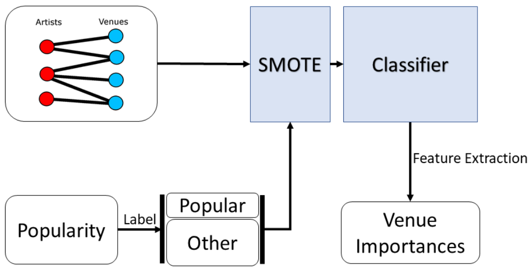


Figure 4: Extracting venue importances.

Classifier

We use the random forest classifier of the *sklearn* Python package [18] with an entropy feature selection criterion. Compared to other classification algorithms, random forests calculate feature importances during the training process and do not require tuning. On the other hand, they can produce lower importances for cross-correlated features. To improve the robustness of such features, we instead deploy an ensemble of random forests [20], which averages importance scores obtained from 10 random forests. To avoid erroneously overstating the importance of unique venue appearances, we train these ensembles and produce importances only for venues in our data where at least 2 artists have performed, which number 602.

Validation

To assert the validity of feature importances assigned by random forest ensembles, we performed leave-one-out cross-validation on trained random forests across 13 training repetitions. For predicting high $popularity$ labels we obtained 9% false positive error rate (i.e. rate of assigning unpopular artists as popular) and 7% false negative error rate (i.e. rate of assigning popular artists as unpopular), whereas for predicting high inc_m labels we obtained 29% false positive error rate and 33% false negative error rate. Since error rates reflect informed classification, venue importances obtained through this process can indeed be considered as the ground truth for subsequent experiments. These error rates indicate that $popularity$ importances are more accurate, although from a methodological standpoint causation is better explored by inc_m importances.

3.3 Compared Ranking Algorithms

In this section, we explore the performance of unsupervised ranking algorithms (such as those presented in Section 2) that aim to rank venues using only Facebook data. These algorithms require prior rank estimations, which we heuristically infer through metadata obtained from the Facebook Graph API. In particular, we estimate artist and venue prior ranks respectively as:

$$a_0 = log(1 + fans + mentions/2)$$
$$b_0 = |\text{events in venues}|$$

[4] To obtain the outliers residing in the right 20% distribution tail, we use the z-score detection threshold 0.84 for those greater than the median.

We alternatively tried calculating venue prior ranks by summing of the size of all events hosted in a venue, heuristically estimated by $size = \max\{0, \log(1 + attending + interested/2 + maybe/2 - noReply/2 - declined)\}$. However, this reduced all BiRank evaluations more than 30% compared to their currently reported values. We compare the following algorithms:

Raw: Estimates venue ranks as their prior ranks.

RFE: Feature extraction using random forest ensembles, similarly to ground truth construction, but aiming to predict high artist prior ranks.

BiRank: BiRank on the artist-venue bipartite graph extracted from Facebook data. Unless stated otherwise, this method uses parameters $r_a = r_v = 0.85$, which are a common empirical selection for ranking algorithms [8, 12].

VenueRank: VenueRank on the artist-venue bipartite graph extracted from Facebook data. As argued above, VenueRank eventually removes the effect of prior ranks. Although inconsequential from a theoretical standpoint, we follow previous conventions and reasoning well-established for BiRank [12], to select the parameters $r_a = r_v = 0.85$ and $p_a = p_v = 0.5$, unless stated otherwise.

Evaluation Measures

To evaluate bipartite venue ranking algorithms, we compare the ranks they produce when applied on Facebook data with the ground truth importances extracted from Spotify data in Subsection 3.2. Our aim is to find whether venues are correctly ranked by unsupervised ranking algorithms. To this end, we measure rank similarities using the robust Spearman correlation coefficient [5], which is is computed as a Pearson correlation between the cardinal ranks of compared quantities. It must be noted that, due to the possibility of negative exposures being found more important, the supremum of Spearman correlation can be less than 1. This, however, does not affect the fact that Spearman correlations closer to 1 indicate that higher ranked venues are more important and thus boast higher exposure.

Additionally, if $ranks_{GT}$ lists venues in a descending order of their ground truth importances and $ranks_C$ in descending order of their calculated ranks, we can define the overlap between the top N venues:

$$overlap(N) = \frac{|ranks_{GT}[0:N] \cap ranks_C[0:N]|}{N}$$

To evaluate the overall $overlap$ curves across all venues, we also measure their Area Under Curve (AUC) [11], which is a fair method of curve comparison. To calculate this area, we perform numerical trapezoid integration of overlaps and normalize the result by dividing it with the width of the horizontal axis. Higher AUC values represent better ability to recognize both high-exposure and low-exposure venues.

3.4 Results

Experiments are performed under two variants of unsupervised training on Facebook data: venue ranking on the same 224 artists (**224A**) (including venues with only one performance) as those used for ground truth construction and venue ranking using all 542 artists (**542A**) and their respective venues. Since the latter dataset contains more artists and venues, it presents a more challenging setting. Using both variants for evaluation helps identify which algorithms generalize better and are more robust.

In Table 2, we can see that BiRank and VenueRank achieve high correlation values with the ground truth in the 224A dataset. However, BiRank heavily relies on accurate prior ranks to do so and does not perform well in the larger 542A dataset, where it produces worse estimations compared to even its prior ranks. This implies that BiRank exhibits overfitting characteristics. On the other hand, both RFE and VenueRank boast great robustness in that they are less affected by the transition to the larger 542A dataset. Consequently, VenueRank exhibits high performance and is more suited to real-world applications, since it is more robust to artist-venue graph changes.

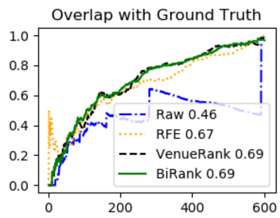
Since there exist -to the best of our knowledge- no previous studies that can serve as comparison for venue correlations, common guidelines [7] suggest that we can resort to the Cohen convention [4] to classify extracted venue ranks as strongly correlated between VenueRank and ground truth importances across all experiments.

	$popularity$		inc_m	
Algorithm	**224A**	**542A**	**224A**	**542A**
Raw	36%	38%	44%	42%
BiRank	70%	33%	76%	28%
RFE	57%	51%	64%	49%
VenueRank	71%	63%	69%	60%

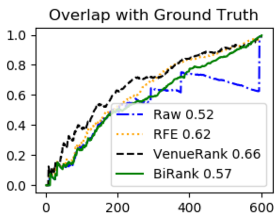
Table 2: Spearman correlation coefficient between ground truth venue rankings and rankings produced by algorithms.

Feature importances forming the $popularity$ and inc_m ground truths are themselves significantly correlated, with 58% Spearman correlation coefficient. This indicates that venues characterizing popular artists also tend to characterize higher listener increments and conversely.

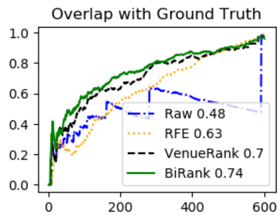
Figure 5 shows the overlap between various algorithms and the ground truth for different numbers of top venues. We can see that, for a small number of top venues (i.e. less than 200), ranking methods do not produce high overlap with ground truth venues. However, for a greater number of venues, they rank highly a large portion of important venues. A curve over a larger area is more important than only identifying top venues, because ranking methods may be used to compare middle-ranked or low-ranked venues to the majority of artists instead of only the most popular ones. AUC results corroborate the previous ones. In particular, BiRank again performs better than other methods under perfect information, whereas VenueRank performs better than other methods and is thus more robust in the case of the more challenging 542A dataset variant.



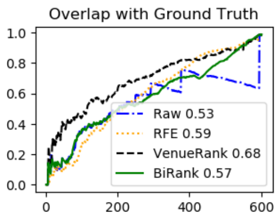
(a) 224A for $popularity$



(b) 542A for $popularity$



(c) 224A for inc_m



(d) 542A for inc_m

Figure 5: Curves and AUCs of high-ranked venue overlap between ranking algorithms and the ground truth.

3.5 Convergence when Ignoring Prior Ranks

In Figure 6 we present the convergence time of BiRank with respect to its iterative scheme parameters r_a, r_b. We can see that execution time increases asymptotically to infinity as prior ranks are ignored, i.e. $r_a = r_b \to 1$. Instead, VenueRank exhibits similar behavior for these parameters convergence-wise, and it always converges to the same stationary solution, as long as these parameters are not close enough to 1 to cause numeric errors. Furthermore, that solution is the same as BiRank when the effect of the prior ranks is completely eliminated. Hence, when the effect of prior ranks is undesireable, it is preferable to employ VenueRank instead of selecting BiRank parameters close to 1, if we want to achieve faster convergence.

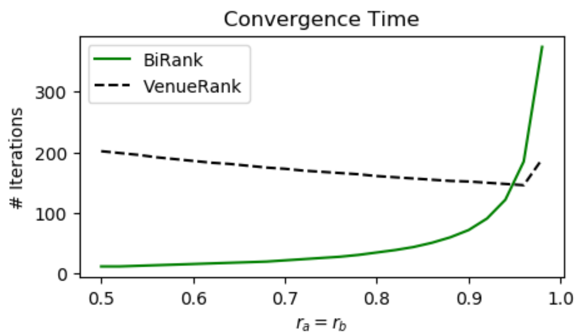


Figure 6: Convergence time of BiRank (green solid line) and VenueRank (dashed black line). VenueRank always has the stationary solution of BiRank with $r_a = r_b \to 1$.

3.6 Case Study

Finally, we conduct a case study, where we try to find important venues in the city of Stockholm, Sweden through venue ranking algorithms. To this end, we used two online articles[5,6] to gather a total of 10 highly recommended venues and find their rankings obtained from running the previous algorithms on all 542 artists and 5,041 venues. In Table 3 we show their rank within the ordered list of all 635 Stockholm venues in our dataset (rank of the highest-ranked venue is 1).

VenueRank places 8 of the 10 venues in the top 50 ranks, whereas other methods place at most 5 of the 10 venues in the top 50 ranks. VenueRank's performance is commendable, given that our dataset also includes popular parks and hotels often used for live music acts, which would not be recommended in the above articles, and that worse ranks often stem from incomplete data (e.g. the dataset contains only two events hosted in 'Nalen').

	Raw	RFE	BiRank	VenueRank
Annexet	95	105	147	40
Berwaldhallen	48	94	96	46
Cirkus	67	61	193	23
Debaser Medis	9	29	9	11
Debaser Rest.	3	8	4	1
Fasching	55	50	125	33
Nalen	195	97	172	113
Pet Sounds Bar	81	19	343	49
Sodra Teatern	39	1	24	2
Stallet	11	63	13	68

Table 3: Rank cardinality for recommended venues compared to other Stockholm venues.

4. CONCLUSIONS AND FUTURE WORK

In this work, we introduce VenueRank as a modification of the common iterative scheme of bipartite graph ranking algorithms that removes dependence on prior ranks while ensuring convergence. We then explore ranking algorithms that help identify which venues help predict artist popularity. Experiments on real-life data show that VenueRank applied on a Facebook artist-venue graph can robustly identify which venues are correlated with more popular artists and actively contribute to increasing their Spotify listeners. In particular, in a setting with partially inaccurate information, VenueRank yields substantial improvement compared to other unsupervised ranking algorithms.

In part, this shows that graph structure can be more important than rough social network metrics when predicting high-exposure venues. Furthermore, it demonstrates that there exists a clear link between graph structure and venue exposure that increases artist popularity.

In the future, we plan to carry out more detailed experiments on larger datasets. Furthermore, from a theoretical perspective, the VenueRank iterative scheme can also be combined with BiRank to produce more robust solutions across the whole parameter space. Finally, we propose improving venues ranks by taking into account how they contribute to the exposure of lower popularity artists.

[5] https://theculturetrip.com/europe/sweden/articles/the-6-best-live-music-venues-in-stockholm

[6] https://scandinaviantraveler.com/en/places/7-best-music-venues-in-stockholm

5. ACKNOWLEDGEMENTS

The authors would like to thank Playground Music Scandinavia for providing artist data. This work is partially funded by the European Commission under the contract number H2020-761634 FuturePulse.

6. REFERENCES

[1] Lei Cao, Jiafeng Guo, and Xueqi Cheng. Bipartite graph based entity ranking for related entity finding. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 130–137. IEEE Computer Society, 2011.

[2] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[3] Zhiyong Cheng and Jialie Shen. Just-for-me: An adaptive personalization system for location-aware social music recommendation. In *Proceedings of international conference on multimedia retrieval*, page 185. ACM, 2014.

[4] Jacob Cohen. Statistical power analysis for the behavioral sciences 2nd edn, 1988.

[5] Christophe Croux and Catherine Dehon. Influence functions of the spearman and kendall correlation measures. *Statistical methods & applications*, 19(4):497–515, 2010.

[6] Hongbo Deng, Michael R Lyu, and Irwin King. A generalized co-hits algorithm and its application to bipartite graphs. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 239–248. ACM, 2009.

[7] Joseph A Durlak. How to select, calculate, and interpret effect sizes. *Journal of pediatric psychology*, 34(9):917–928, 2009.

[8] Nadav Eiron, Kevin S McCurley, and John A Tomlin. Ranking the web frontier. In *Proceedings of the 13th international conference on World Wide Web*, pages 309–318. ACM, 2004.

[9] Amit Goyal, Francesco Bonchi, and Laks VS Lakshmanan. Learning influence probabilities in social networks. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 241–250. ACM, 2010.

[10] Isabelle Guyon and André Elisseeff. An introduction to feature extraction. In *Feature extraction*, pages 1–25. Springer, 2006.

[11] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[12] Xiangnan He, Ming Gao, Min-Yen Kan, and Dingxian Wang. Birank: Towards ranking on bipartite graphs. *IEEE Transactions on Knowledge and Data Engineering*, 29(1):57–71, 2017.

[13] Boris Iglewicz and David Hoaglin. *Volume 16: how to detect and handle outliers, The ASQC basic references in quality control: statistical techniques, Edward F. Mykytka*. PhD thesis, Ph. D., Editor, 1993.

[14] Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[15] Joel C Miller, Gregory Rae, Fred Schaefer, Lesley A Ward, Thomas LoFaro, and Ayman Farahat. Modifications of kleinberg's hits algorithm using matrix exponentiation and web log records. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 444–445. ACM, 2001.

[16] Erkki Oja and Juha Karhunen. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *Journal of mathematical analysis and applications*, 106(1):69–84, 1985.

[17] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[19] Xiaoguang Rui, Mingjing Li, Zhiwei Li, Wei-Ying Ma, and Nenghai Yu. Bipartite graph reinforcement model for web image annotation. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 585–594. ACM, 2007.

[20] Yvan Saeys, Thomas Abeel, and Yves Van de Peer. Robust feature selection using ensemble feature selection techniques. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 313–325. Springer, 2008.

[21] Markus Schedl and Dominik Schnitzer. Location-aware music artist recommendation. In *International Conference on Multimedia Modeling*, pages 205–213. Springer, 2014.