Mutual proximity graphs for music recommendation

Arthur Flexer[1]* and Jeff Stevens[2]

[1] Austrian Research Institute for Artificial Intelligence
Freyung 6/6, Vienna, Austria
[2] George Mason University, 4400 University Dr, Fairfax, VA, USA

Abstract. We present mutual proximity graphs, which are an extension of mutual *k*-nearest neighbor (knn) graphs, and are able to avoid hub vertices having abnormally high connectivity. We apply this new approach in a music recommendation system based on an incrementally constructed knn graph. We show that mutual proximity graphs yield much better connected graphs with better reachability compared to knn graphs and mutual knn graphs.

1 Introduction

In graph theory, Ozaki et al [4] have observed that standard knn graphs often produce hubs, i.e. vertices with extremely high numbers of edges to other vertices. The same authors already made the connection to the phenomenon of hubness, which is a general problem of learning in high-dimensional spaces which has been discovered in music information retrieval (MIR) [1], but then gained attention in a general machine learning context where it has been discussed as a new aspect of the curse of dimensionality [5, 6]. It has been shown [5] that for any finite dimensionality, some points in a data set are expected to be closer to the center of all data than other points and are at the same time closer, on average, to all other points. Such points closer to the center have a high probability of being hubs, i.e. of appearing in nearest neighbor lists of many other points. Points which are further away from the center have a high probability of being anti-hubs, i.e. points that never appear in any nearest neighbor list.

This hubness phenomenon also negatively impacts a real world music recommendation system which has been built by our research team. This system uses an iteratively constructed knn graph to recommend music via a web interface. In previous work [3, 2] we were able to show that hubness causes some songs to never occur in the nearest neighbor lists at all, since hubs crowd the nearest neighbors lists and are being recommended repeatedly. As a result only about two thirds of the songs are reachable in the recommendation interface, i.e. over a third of the songs are never recommended. Further analysis of the knn graph

* This research is supported by the Austrian Science Fund (FWF–P27082).

shows that only less than a third of the songs are likely to be recommended, since only those are part of one large strongly connected subgraph. We have already applied 'mutual proximity' [6], a hubness reduction method, to improve this situation.

This paper adds an analysis of the properties of the resulting nearest neighbor graph and a comparison to other graph construction methods. We show that application of mutual proximity leads to the formulation of mutual proximity graphs, which can be seen as an extension of mutual knn graphs. Analysis of the respective graphs shows that mutual proximity graphs yield better connected graphs with greater reachability when compared to knn or mutual knn graphs.

2 Data

For our analysis we use data from the real world music discovery system FM4 Soundpark (`http://fm4.orf.at/soundpark`), which is a web platform run by the Austrian public radio station FM4, where artists can upload and present their music free of charge. Visitors of the web site can listen to and download all the music at no cost, with most recent uploads being displayed at the top of the website. To allow a more intuitive and appealing access to the full database regardless of publication date of a song, we implemented a recommendation system using a content-based music similarity measure [3]. This similarity measure is based on timbre information computed from the audio by dividing every track into overlapping frames for which 20 MFCCs are being computed which are modeled via a single Gaussian with full covariance matrix. To compute a distance value between two Gaussians the symmetrized Kullback-Leibler (SKL) divergence is used (see [3] for details on both MFCCs and SKL). This results in an $N \times N$ distance matrix D for the data set, with $N = 7665$ songs. Each of the songs belongs to one or two genres, hence the following percentages add up to more than 100%: 37.6% of all songs belong to genre Pop, 46.3% to Rock, 44.0% to Electronica, 14.3% to Hip-Hop, 19.7% to Funk, 5.3% to Reggae (see [2] for more detail concerning the data set).

3 Methods

The user interface of the music recommender has been implemented as a visualization of an incrementally constructed knn graph showing the five most similar songs to the currently playing one. In what follows we will describe the construction of the knn graph, two alternative construction methods, and measures to evaluate these graphs. A graph $G = (V, E)$ is defined via a finite set of vertices v and edges e. In our case the vertices correspond to songs displayed in the music recommender interface.

***k*-nearest neighbor graphs (knn):** knn graphs are a very standard graph construction technique, where an edge e_{ij} is placed between v_i and v_j if v_j is among the k nearest neighbors of v_i. We use the distance matrix D defined in Sec. 2 to compute an adjacency matrix A. If, according to D, a song with the index j is among the five nearest neighbors of a song with index i, then $A_{ij} = 1$, otherwise $A_{ij} = 0$. An edge e_{ij} exists between two vertices v_i and v_j if $A_{ij} = 1$. Please note that the adjacency matrix A is not symmetric and the resulting graph is therefore a directed knn graph.

Mutual k-nearest neighbor graphs (muknn): muknn graphs (see e.g. [4]) are an extension of knn graphs, where an edge e_{ij} exists only if $A_{ij} = A_{ji} = 1$, i.e. if song j is among the nearest neighbors of i and vice versa. The resulting muknn graph is a subset of the corresponding knn graph, containing a subset of the vertices v, and versions of it have already been applied to reduce hubness [4].

Mutual proximity graphs (mp): for construction of mp graphs we first rescale the distance matrix D using the hubness reduction method mutual proximity (MP) [6]. MP rescales the original distance space so that two objects sharing similar nearest neighbors are more closely tied to each other, while two objects with dissimilar neighborhoods are repelled from each other. MP reinterprets the distance of two objects as a mutual proximity in terms of their distribution of distances. To compute MP, we assume that the distances $D_{x,h=1..N}$ from an object x to all other objects in our data set follow a certain probability distribution, thus any distance $D_{x,y}$ can be reinterpreted as the probability of y being the nearest neighbor of x, given their distance $D_{x,y}$ and the probability distribution $P(X)$. MP is then defined as the probability that y is the nearest neighbor of x given $P(X)$ and x is the nearest neighbor of y given $P(Y)$:

$$MP(D_{x,y}) = P(X > D_{x,y} \cap Y > D_{y,x}). \tag{1}$$

To compute MP in our experiments we use the empirical distribution. Changing from similarities to distances, computation of $1-MP$ yields a so-called secondary distance matrix D^{MP}, which is then used to construct an adjacency matrix A^{MP}. Using A^{MP} we construct a knn graph exactly as described above, which we now call mp graph, since it is based on distances rescaled via MP.

Number of hubs and anti-hubs (#hub, #anti): as a measure of the hubness of a given song we use the number of times the song occurs in the first n nearest neighbors of all the other songs in the data base (so-called n-occurrence). The mean n-occurrence across all songs in a data base is equal to n. Any n-occurrence significantly bigger than n therefore indicates existence of a hub. Since our music recommender always shows the five most similar songs we use $n = 5$. We compute the number of songs of which the n-occurrence is more than five times n (#hub), and for which it is equal zero indicating an anti-hub (#anti).

Reachability (reach): This is the percentage of songs from the whole data base that are part of at least one of the recommendation lists, i.e. $(1 - (\#anti/N)) \times 100$. If a song is not part of any of the recommendation lists of size $n = 5$ it cannot be reached using our recommendation function.

Strongly connected Component (scc, #scc, s$\bar{\text{c}}$c): For our incrementally constructed nearest neighbor graph, a strongly connected component (SCC) is a subgraph where every song is connected to all other songs traveling along the directed nearest neighbor connections. We use Tarjan's algorithm [7] to find all SCC-graphs in our nearest neighbor graph with $n = 5$. We report the size of the largest strongly connected component as a percentage of the whole data base (scc), the number of additional strongly connected components (#scc) and their average size ($s\bar{c}c$) in number of vertices. Please note that sccs can have significant vertex overlap, i.e. sccs are not disjoint sets of vertices.

Number of edges (#edges): we give the number of edges e in a graph G.

ϕ-edge ratio (ϕ): for a labeled graph (G, l) a ϕ-edge is any edge e_{ij} for which $l_i \neq l_j$ [4], i.e. in our case for which the genres l of the songs corresponding to the vertices v_i and v_j do not match. Since our vertices can have one or two labels (genres), each ϕ-edge is assigned a value according to the percentage of non-overlapping genres between vertices. The ϕ-edge ratio is the sum of ϕ-edge values divided by the total number of edges, i.e. the percentage of edges connecting vertices with different labels. If the edges in a graph reflect the semantic meaning of its labeled vertices, it will have a low ϕ-edge ratio.

4 Results

Our analysis results using the evaluation indices defined in Sec. 3 are given in Table 1. The knn graph, which is actually being used in the FM4 Soundpark, shows considerable hubness. The number of hubs ($\#hub$) is 291 and there are 2661 anti-hubs ($\#anti$). As a consequence only 65.28% of vertices are reachable at all. About a third of the data is therefore never being recommended. Looking at the muknn graph, all hubs are gone but there is a high number of anti-hubs (4566). Hubs vanish as a consequence of the strict requirement of mutual neighborhoods when building the muknn graph, since now every vertex can only be connected to at most five other vertices. This also has the consequence that many edges from the knn graph are being deleted ($\#edges = 5790$ instead of $7665 \times 5 = 38325$ for knn and mp), which produces a very high number of anti-hubs (4566), and a low reachability of 40.43%. The mp graph shows a low number of 2 hubs and a moderate number of 642 anti-hubs, with a highly increased reachability of 91.62%. Looking at the size of the largest strongly connected component (scc), one can see that it contains only 29.11% of the vertices for knn, which further decreases to 11.89% for muknn. Aside from this largest scc, there exist large numbers of additional sccs ($\#scc$) of very small average size ($s\bar{c}c$) for both knn (409 with average size of 2.87) and muknn (653 with average size of 4.75). This means that it is very likely that a user of the recommendation system will spend most of their time listening to songs within the largest scc, which comprises only one third of the data for knn, or just about one tenth for muknn. The scc for the mp graph on the other hand is much larger, comprised of 85.26% of the vertices, with 103 extra sccs of average size 66.29. It seems that due to hubness reduction, many vertices now connect to others instead of hubs, so the mp graph exhibits a much higher connectivity than the original knn graph. The muknn graph shows absolutely no hubness, but deletion of many edges during construction of the graph yields a very disconnected and not very useful graph. It has already been suggested [4] to add a minimum spanning tree to the muknn graph to ensure better connectivity. It will be interesting to compare such an enriched muknn graph to the mp graph.

Table 1. Graph analysis of knn, muknn and mp graphs.

graph	#hub	#anti	reach	scc	#scc	$s\bar{c}c$	#edges	ϕ
knn	291	2661	65.28%	29.11%	409	2.87	38325	55.78%
muknn	0	4566	40.43%	11.89%	653	4.75	5790	37.79%
mp	2	641	91.62%	85.26%	103	66.29	38325	52.20%

Finally, looking at the percentage of edges connecting vertices with different labels (ϕ), we can see that both muknn and mp yield improvements compared to knn. Whereas the ϕ-edge ratio is 55.75% for knn, this improves to 37.79% for muknn and slightly improves to 52.20% for mp. This means that the hubness reduction in the muknn and mp graphs also respects the semantic meaning of the data when deviating from the original knn graph.

5 Conclusion

We have presented mutual proximity graphs, which are a new type of nearest neighbor graph able to decisively reduce hub vertices with extremely high numbers of edges. Whereas the related mutual k-nearest neighbor graphs are able to completely prevent formation of hub vertices, they result in graphs with not very useful overall connectivity. Whereas mutual knn graphs have the strict requirement of connecting only vertices which belong to each other's nearest neighbors, mutual proximity graphs do this in a more flexible probabilistic way. We showed that mutual proximity graphs can improve a real world music recommendation system, but future work should also explore usefulness of this new approach in other scenarios where hub vertices can be found.

References

1. Aucouturier, J.-J., Pachet F.: Improving Timbre Similarity: How high is the sky?, Journal of Negative Results in Speech and Audio Sciences, 1(1), pp. 1-13, 2004.
2. Flexer A., Gasser M., Schnitzer D.: Limitations of interactive music recommendation based on audio content, Proc. of the 5th Audio Mostly Conf. pp. 96-102, 2010.
3. Gasser M., Flexer A.: FM4 Soundpark: Audio-based Music Recommendation in Everyday Use, in Proc. of the 6th Sound and Music Computing Conf. pp. 161-166, 2009.
4. Ozaki K., Shimbo M., Komachi M., Matsumoto Y.: Using the mutual k-nearest neighbor graphs for semi-supervised classification of natural language data, in Proc. of the 15th Conf. on Computational Natural Language Learning, pp. 154-162, 2011.
5. Radovanović M., Nanopoulos A., Ivanović M.: Hubs in space: Popular nearest neighbors in high-dimensional data, Journal of Machine Learning Research, 11:2487-2531, 2010.
6. Schnitzer D., Flexer A., Schedl M., Widmer G.: Local and Global Scaling Reduce Hubs in Space, Journal of Machine Learning Research, 13(Oct):2871-2902, 2012.
7. Tarjan R.: Depth-first search and linear graph algorithms, SIAM Journal on Computing, Vol. 1, No. 2, pp. 146-160, 1972.